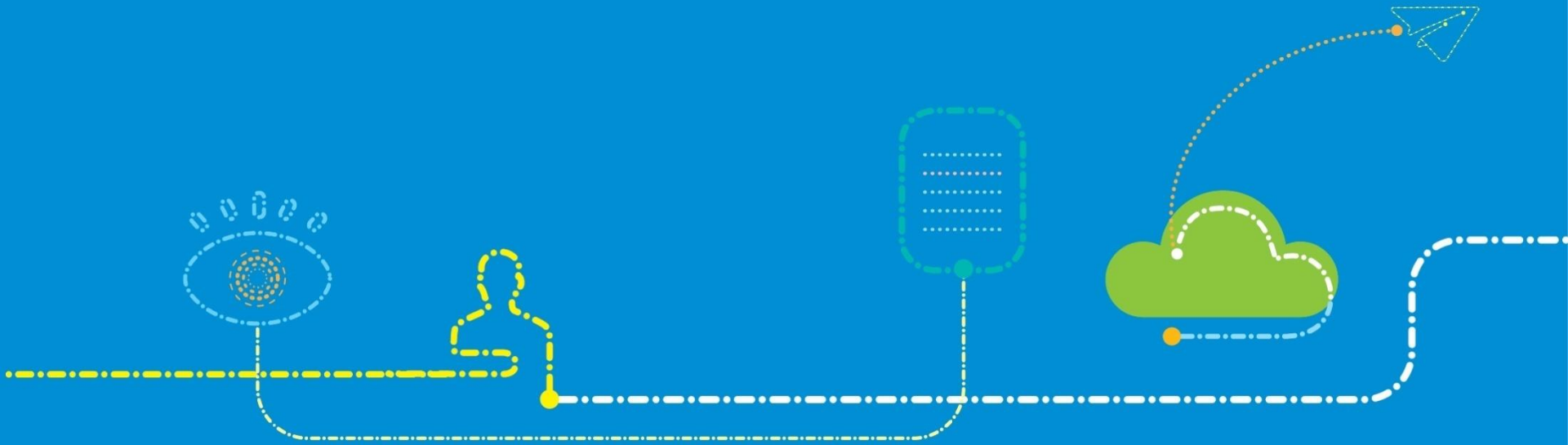
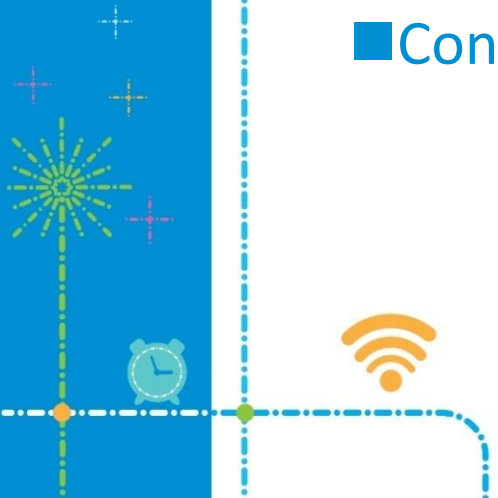
Building The Networking Infrastructure for MEC & IoT Based On FD.io/VPP



Agenda

- MEC & IoT Network Service Framework
- use case 1 : MEC Traffic Offloading Locally
- use case 2: IoT Gateway
- Contributions for VPP



➢ Problem statement

The development of mobile communications, such as VR/AR, Connected vehicles, Industrial control, IoT and so on

- Higher requirement for high bandwidth, ultra-low latency
- The load of core network is further aggravated

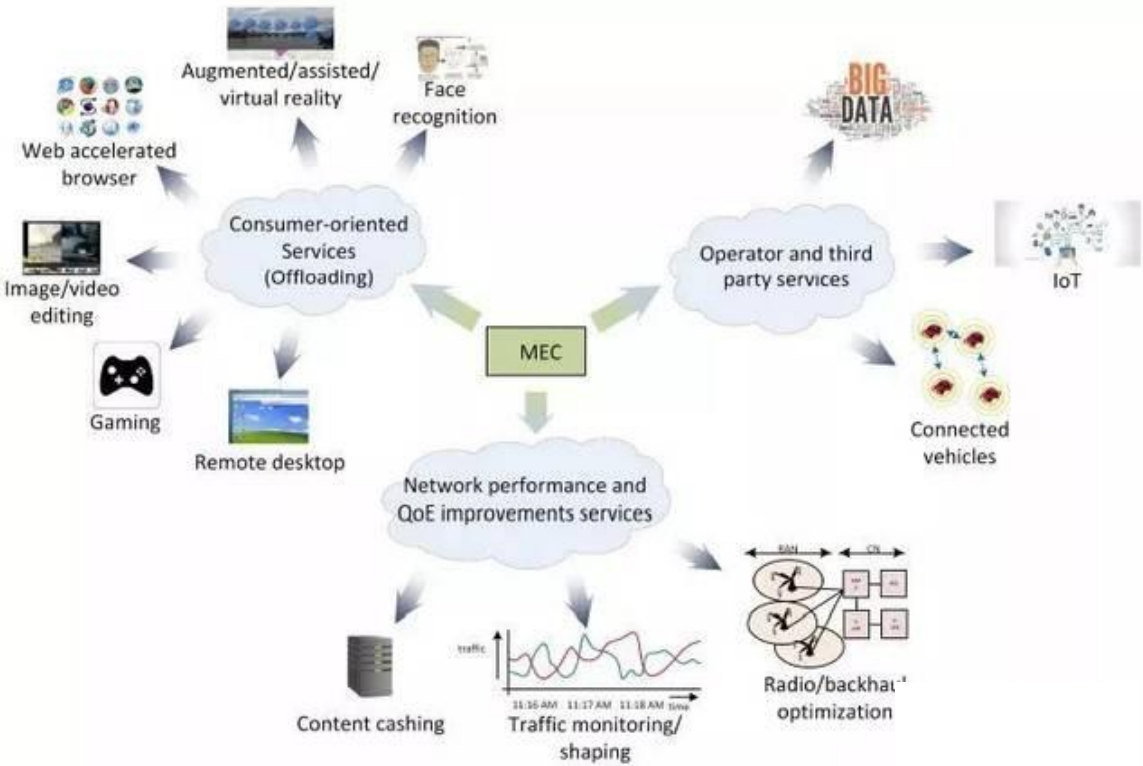
➢ MEC proposed

- ETSI proposed MEC, based on 5G evolution architecture
- Providing IT services and cloud-computing capabilities at the edge of the mobile network

MEC&IoT

➢ Scenarios

- Intelligent video acceleration
- Video stream analysis
- Augmented reality
- Assistance for intensive compute
- Enterprise deployment of MEC
- Connected vehicles
- IoT gateway

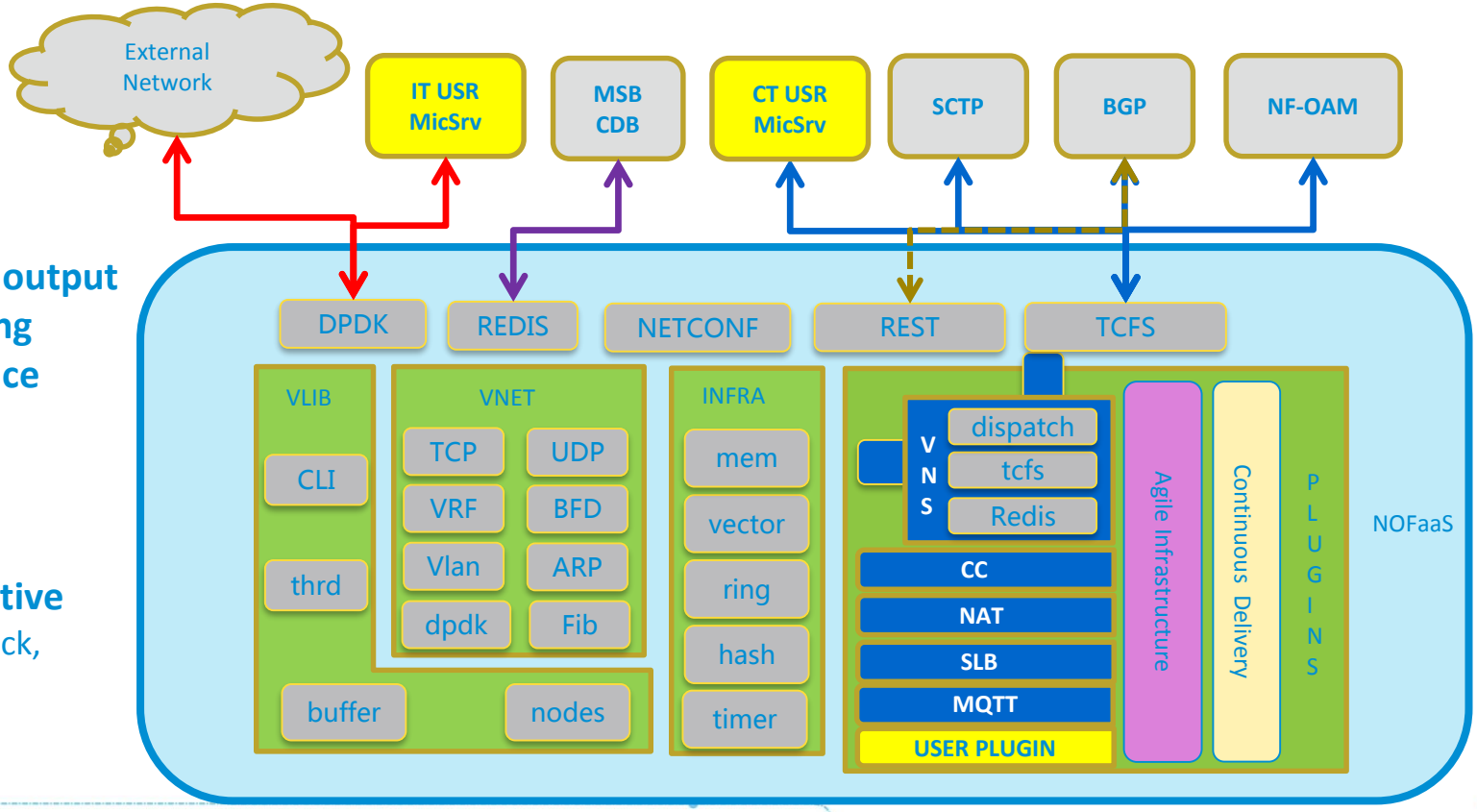


MEC&IoT Network Characteristic	FD.io Advantage
High Throughput	VPP & DPDK
Low Latency	
Huge Amounts of Link	High-Performance Bi-hash Table
Service Localized	Plugin as a node
Traffic offloading	High-Performance L2/L3 Forwarding
Flexible Service	Flexible Deployment at VM/Container

ZTE-NOFaaS Cloud Native Framework

Network output forwarding as a service

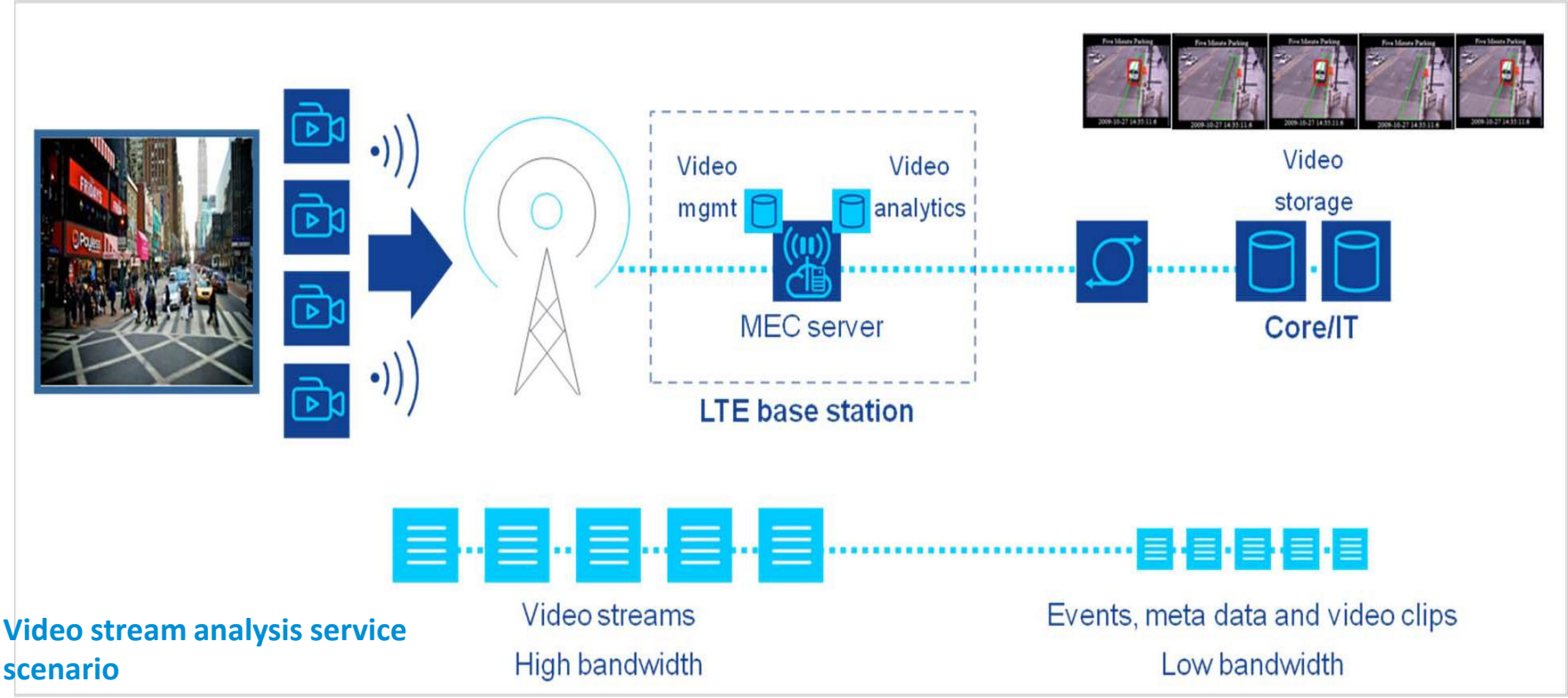
Cloud Native
CI, KW check, UT,FT



Agenda

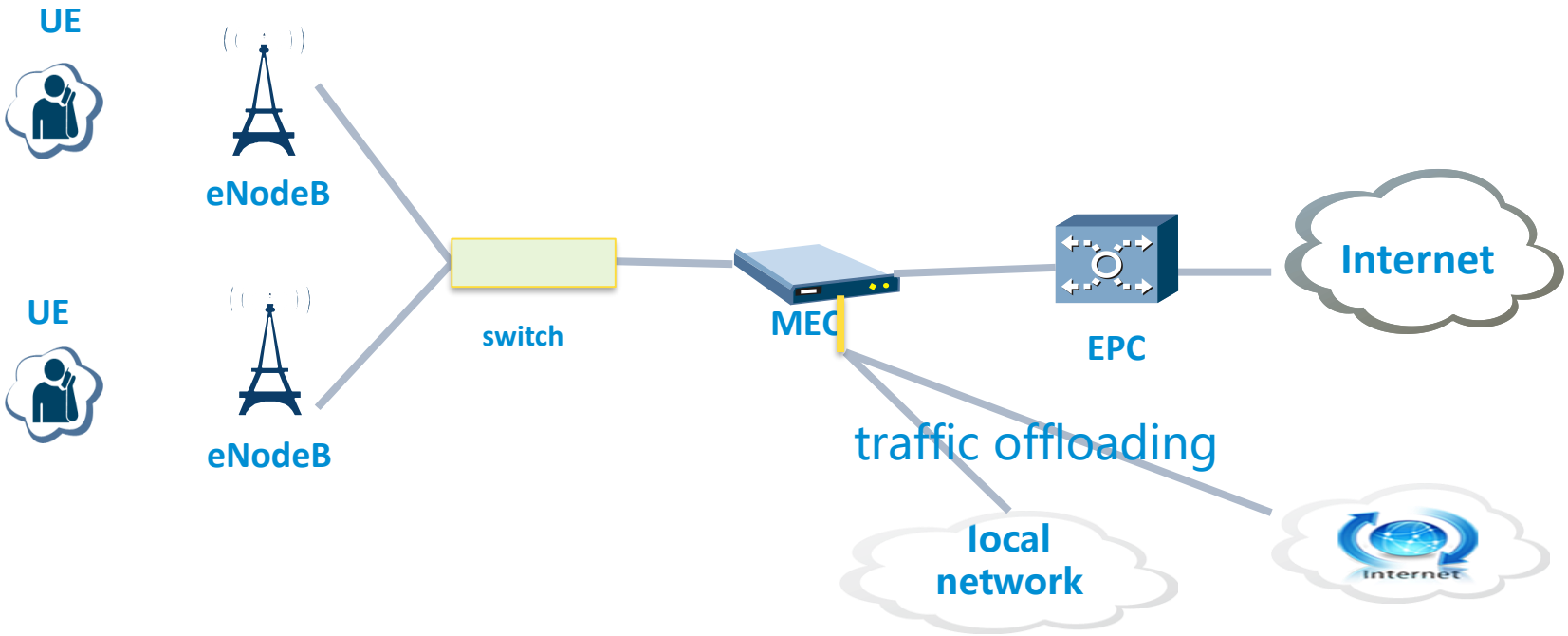
- MEC & IoT Network Output Forwarding Framework
- **use case 1 : MEC Traffic Offloading Locally**
- use case 2: IoT Gateway
- Contributions for VPP

MEC -Application Scenarios(Video stream analysis)

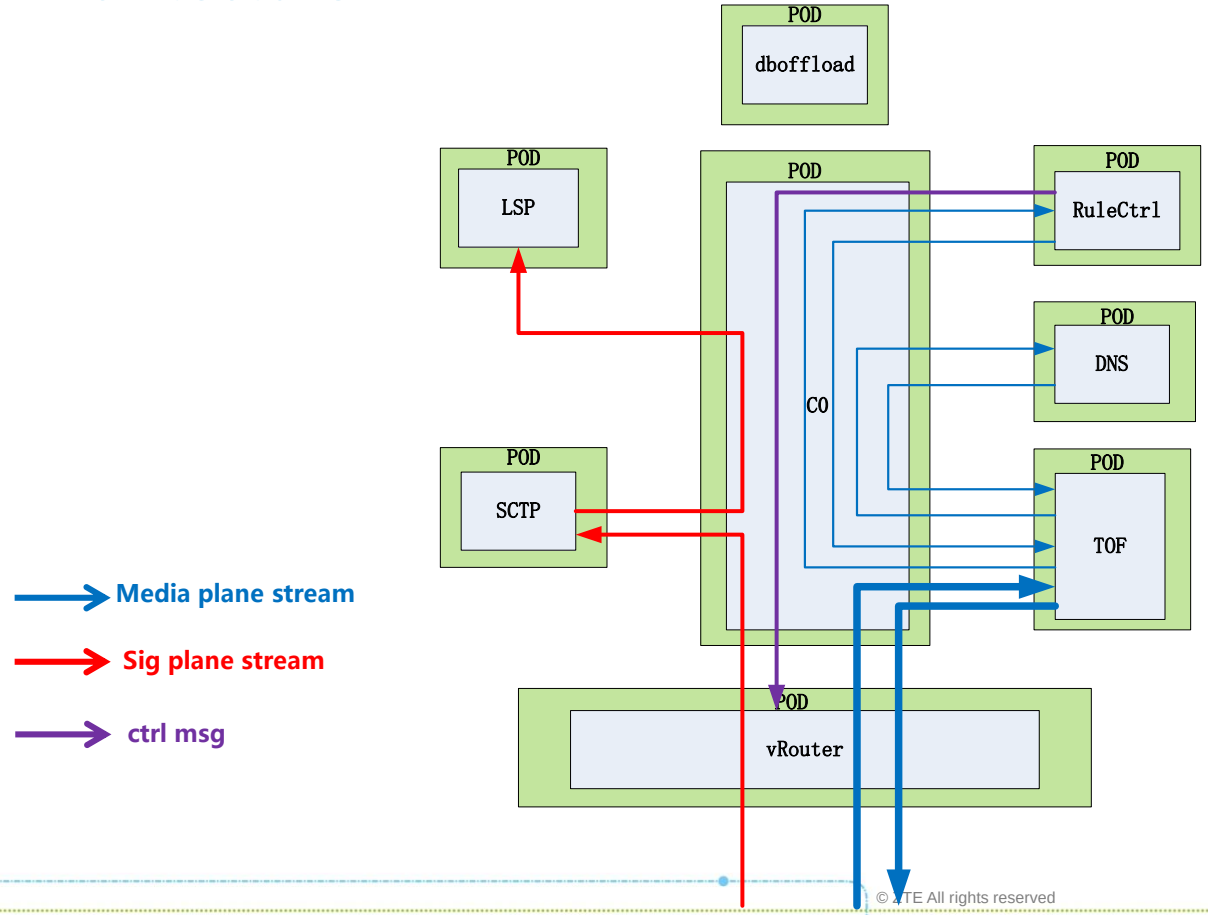


Video stream analysis service scenario

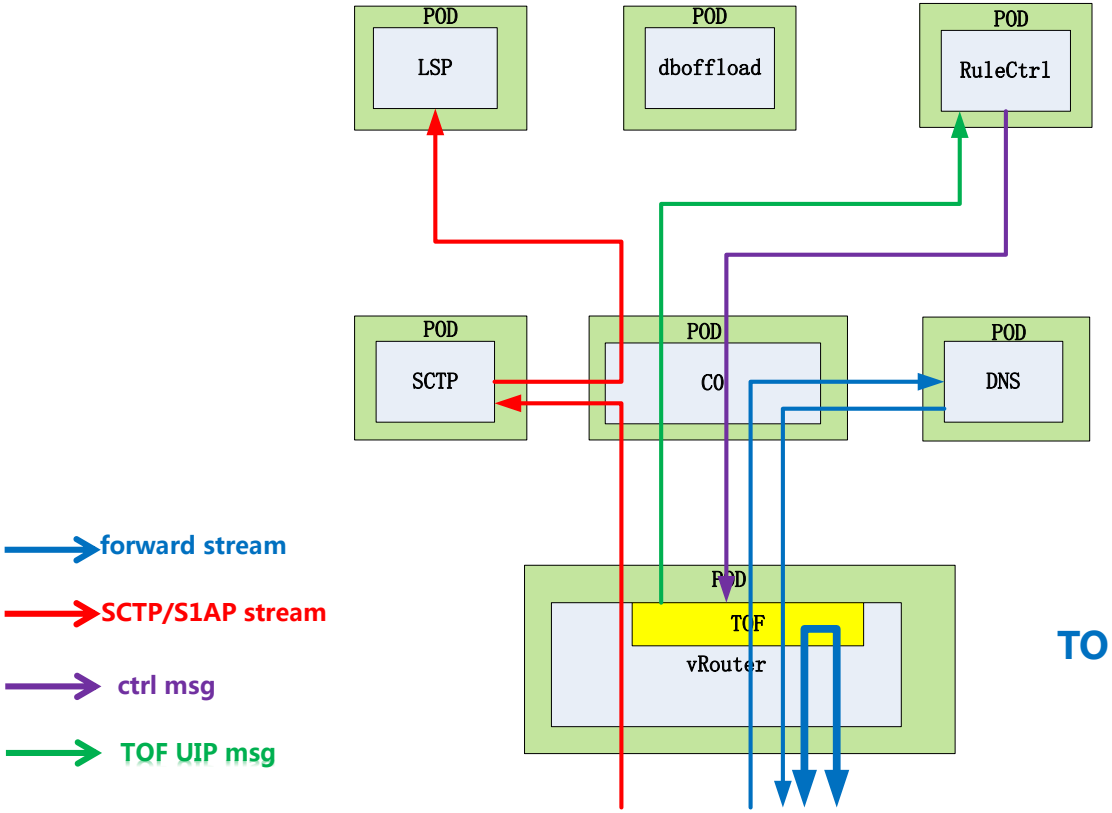
MEC-Network topology



MEC-Architecture



MEC-Architecture2



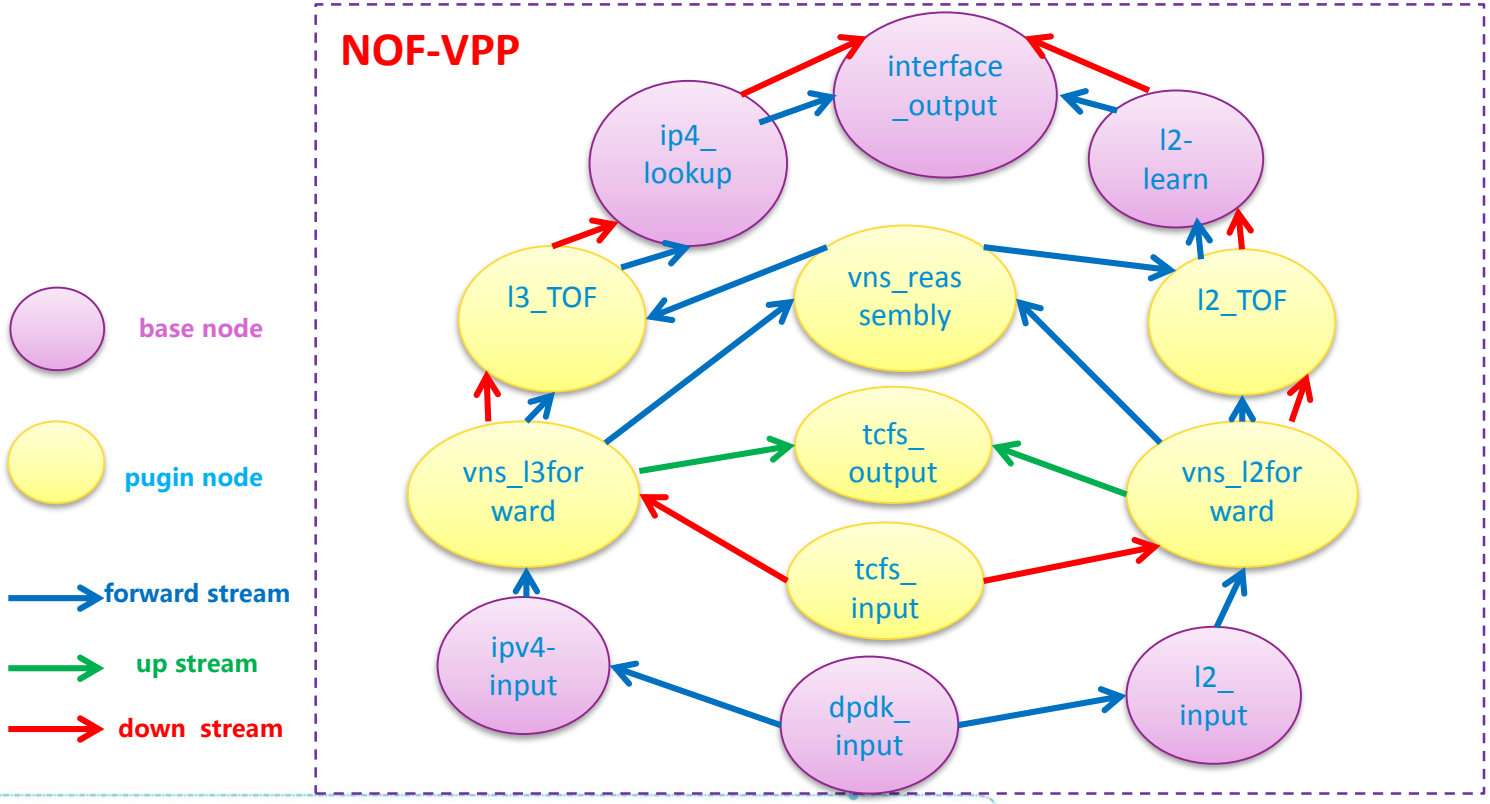
TOF run in VPP as a node

MEC-VPP key technology

Improvement

- classifier table -Classifier tabel dynamic generation
- ARP table - Free Arp、 Arp aging、 No Arp
- Fragment/Reassembly - Ip packet Pseudo Reassembly
- NAT - full NAT

MEC-Node Graph Rearrange



MEC-L3、 L2 Forwading Performance Test Teport

Hardware ： 5300 server， 2 Cores， 2x10ge NICs

-CPU： Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz

-Memery： Samsung DDR4 128G @ 2133 MHz

-Eth-NIC: Intel 82599 10 Gigabit X2

-container

L3

Frame Size (bytes)	Intended Load (%)	Offered Load (%)	Throughput (%)	Aggregated Throughput (fps)	Aggregated Theoretical Max (fps)
64	39.531	39.26	39.26	11,684,401.1	29,761,904.762
128	69.063	68.519	68.519	11,574,074.1	16,891,891.892
256	99.297	98.571	98.571	8,928,571.433	9,057,971.014
512	100	100	100	4,699,248.133	4,699,248.12
1,024	100	100	100	2,394,636.033	2,394,636.015
1,280	100	100	100	1,923,076.933	1,923,076.923
1,518	100	99.87	99.87	1,623,376.633	1,625,487.646

MEC-L3、 L2 Forwading Performance Test Teport

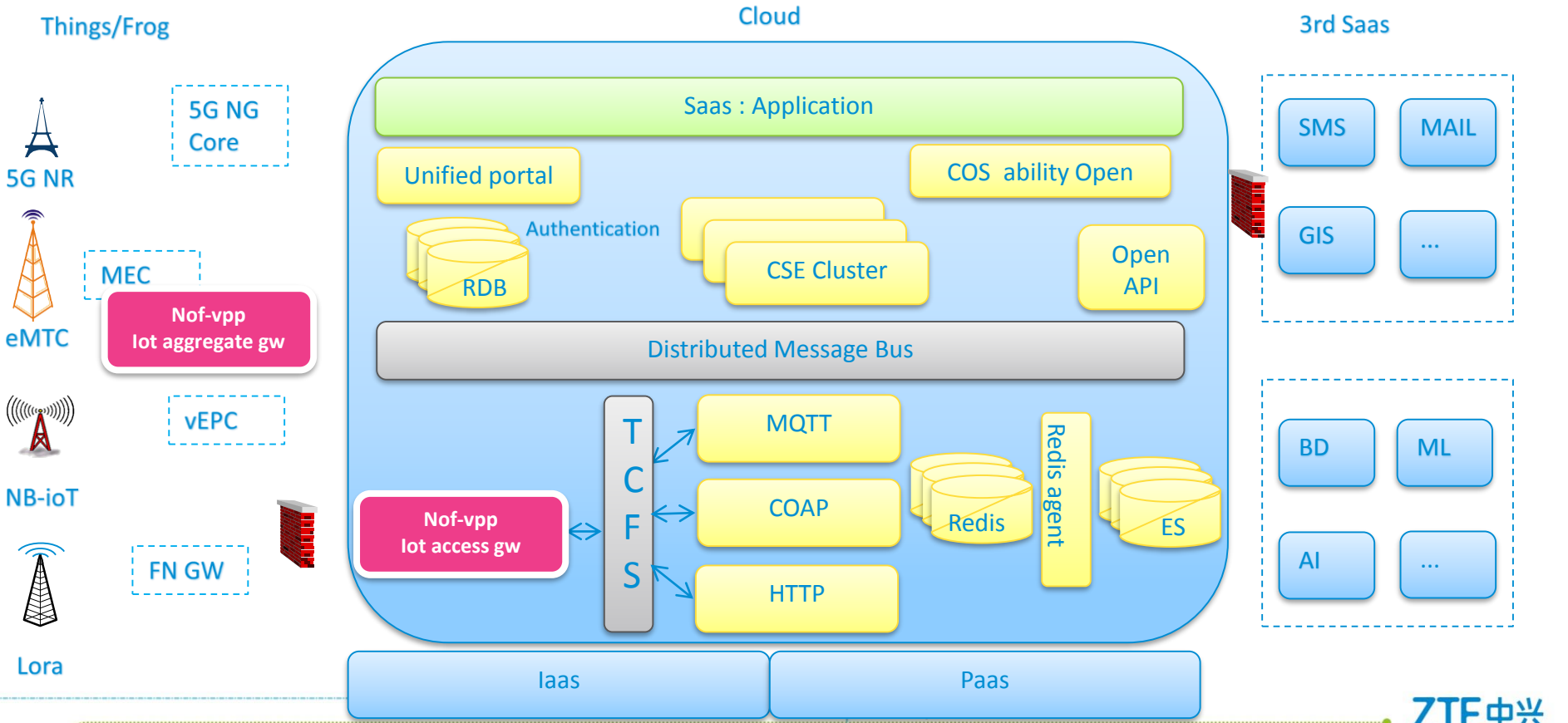
L2

Frame Size (bytes)	Intended Load (%)	Offered Load (%)	Throughput (%)	Aggregated Throughput (fps)	Aggregated Theoretical Max (fps)
64	49.375	48.837	48.837	14,534,883.733	29,761,904.762
128	87.344	86.047	86.047	14,534,883.733	16,891,891.892
256	99.297	98.571	98.571	8,928,571.433	9,057,971.014
512	100	100	100	4,699,248.133	4,699,248.12
1,024	100	100	100	2,394,636.033	2,394,636.015
1,280	100	100	100	1,923,076.933	1,923,076.923
1,518	100	99.87	99.87	1,623,376.633	1,625,487.646

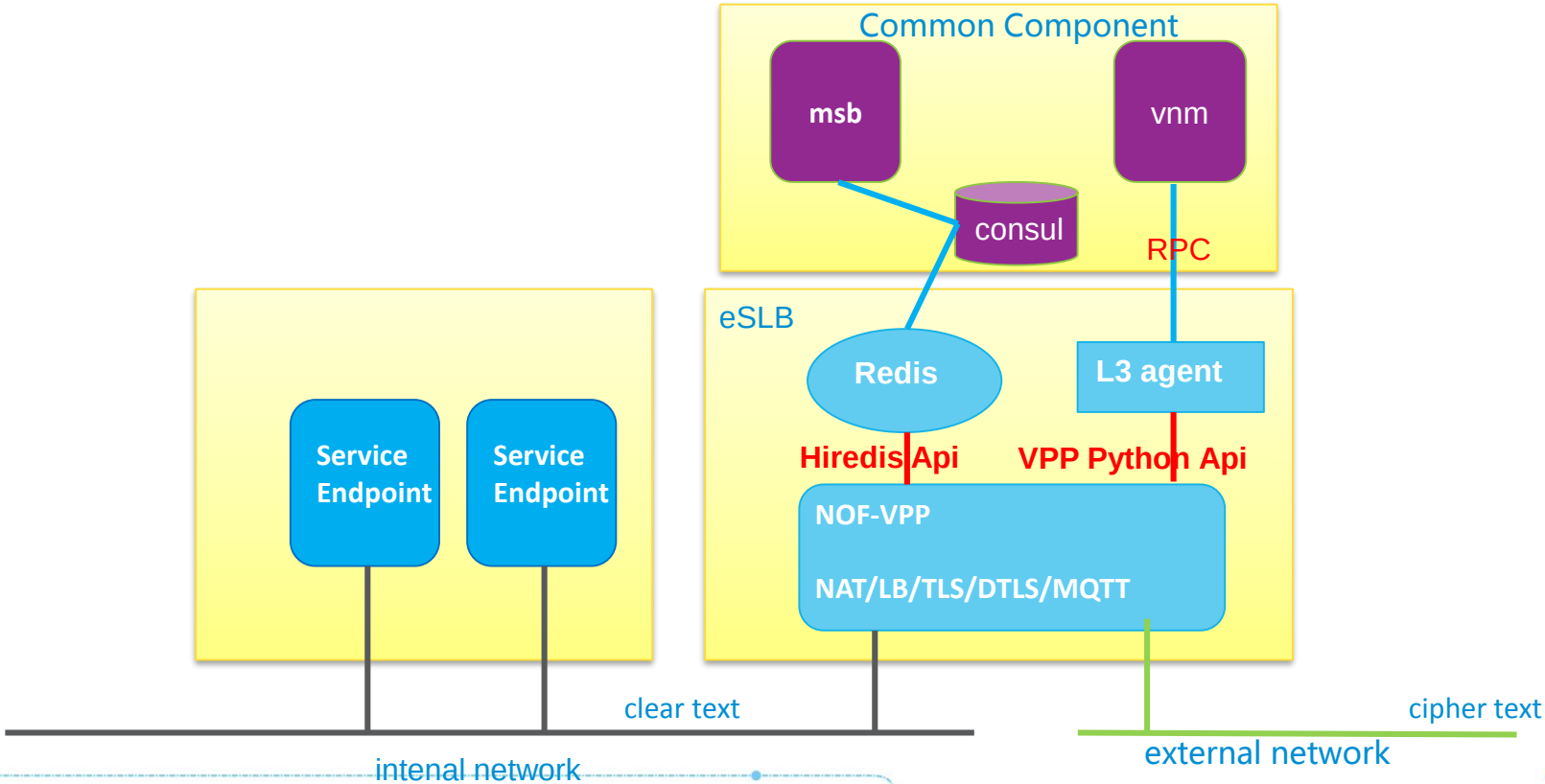
Agenda

- MEC & IoT Network Output Forwarding Framework
- use case 1 : MEC Traffic Offloading Locally
- **use case 2: IoT Gateway**
- Contributions for VPP

IoT gateway - Application Scenarios

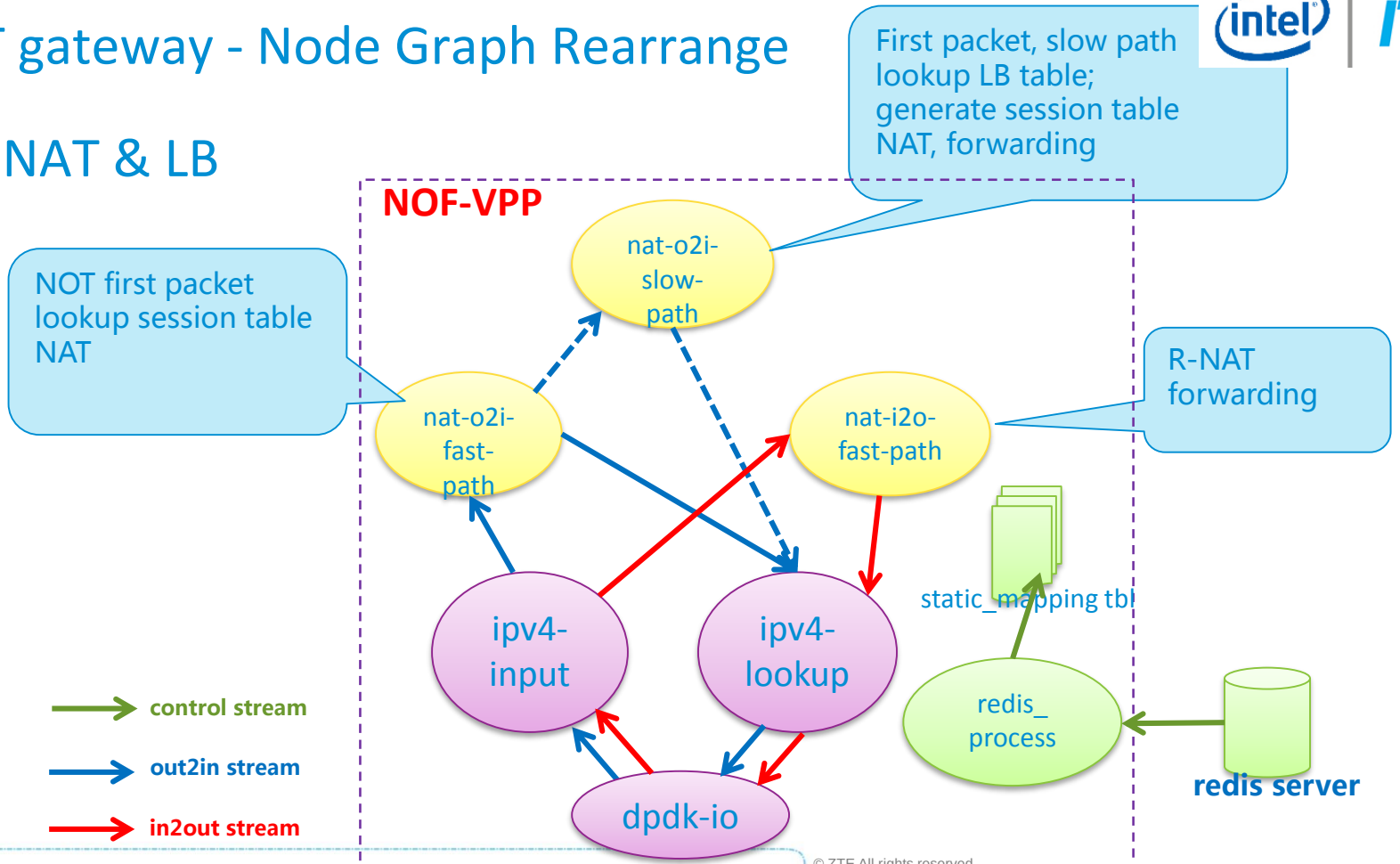


IoT gateway - Architecture



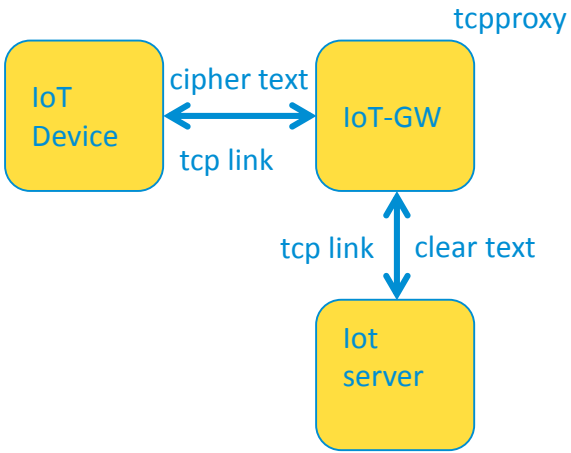
IoT gateway - Node Graph Rearrange

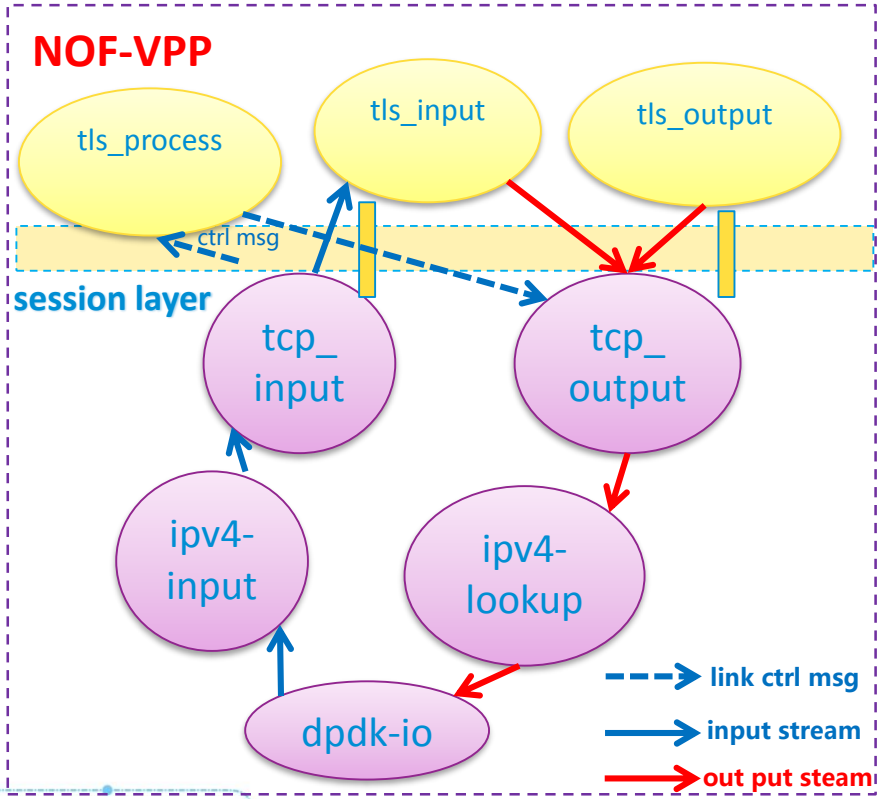
➢ NAT & LB



IoT gateway - Node Graph Rearrange

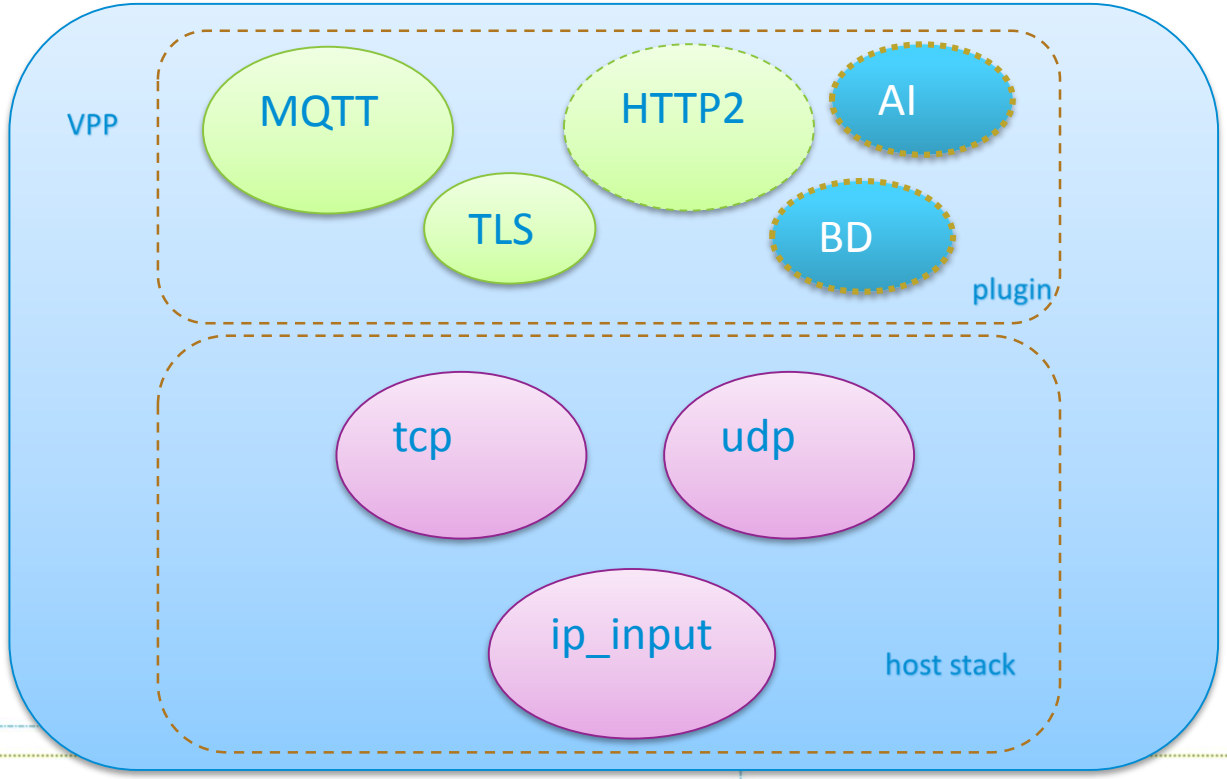
➢ TLS/DTLS





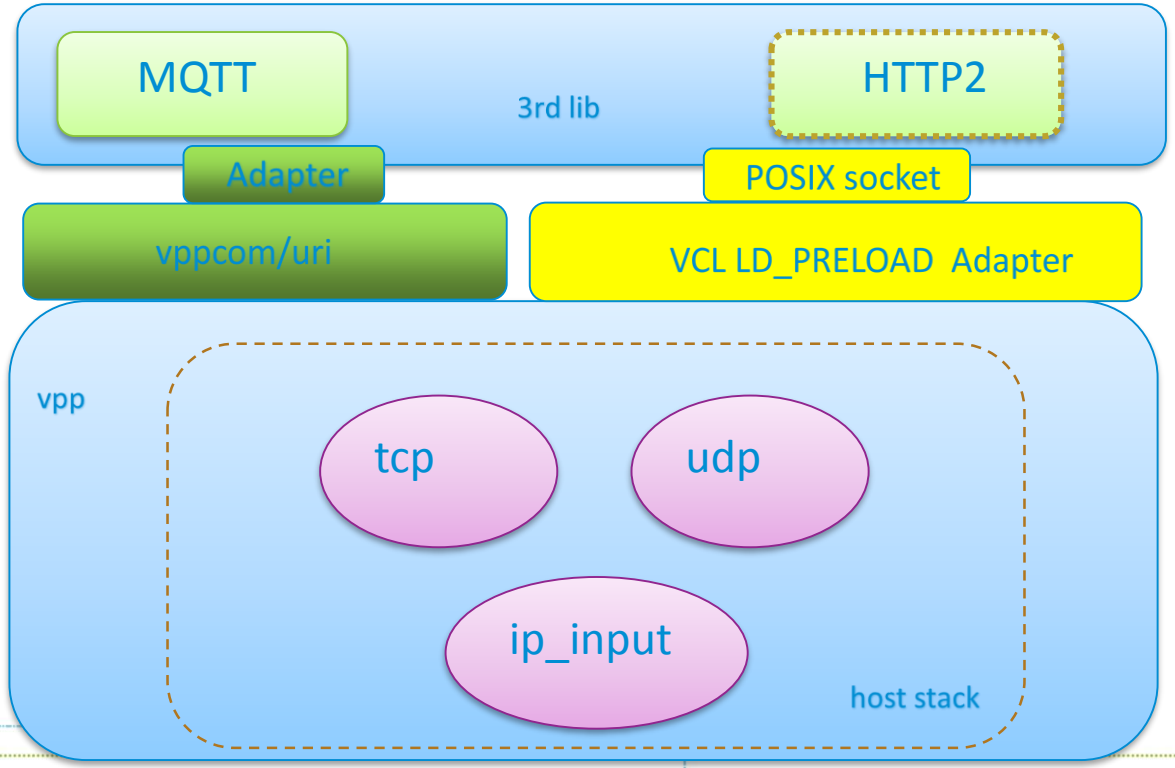
IoT gateway - Node Graph Rearrange

- L7 protocol or APP - Builtin



IoT gateway - Node Graph Rearrange

- L7 protocol or APP - VCL and LD_PRELOAD



IoT gateway - NAT Performance Test Teport

Hardware ： 5300 server， 1 Cores， 2x10ge NICs

-CPU： Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz

-Memery： Samsung DDR4 128G @ 2133 MHz

-Eth-NIC: Intel 82599 10 Gigabit X2

NAT&LB out2in

Frame Size	Intended Load (%)	Offered Load (%)	Throughput (%)	Offered Load (fps)	Forwarding Rate (fps)
64	29.688	29.577	29.577	4401408.467	4401405.73
128	52.891	52.857	52.857	4464285.717	4464282.52
256	99.297	98.571	98.571	4464285.717	4464280.27

Contributions for VPP

➢ Commercial Functions Improvement

ARP module improvement

IP fragment packet pseudo reassembly improvement

NAT ip fragment & icmp fragment packet pseudo reassembly support

LB static scheduling algorithm (rr,wrr,hash)supplement;LB service health check support

➢ Maintainable

Segment fault invocation chain showout; Packet trace improvement

➢ Other

Compile the VPP framework into lib

谢谢！

未来，不等待