Use Cases of CAN

draft-liu-dyncast-ps-usecases-03

P. Liu, China Mobile
P. Eardley, British Telecom
D. Trossen, Huawei
M. Boucadair, Orange
LM. Contreras, Telefonica
C.Li,Huawei

CAN IETF: Problem domain for IETF routing

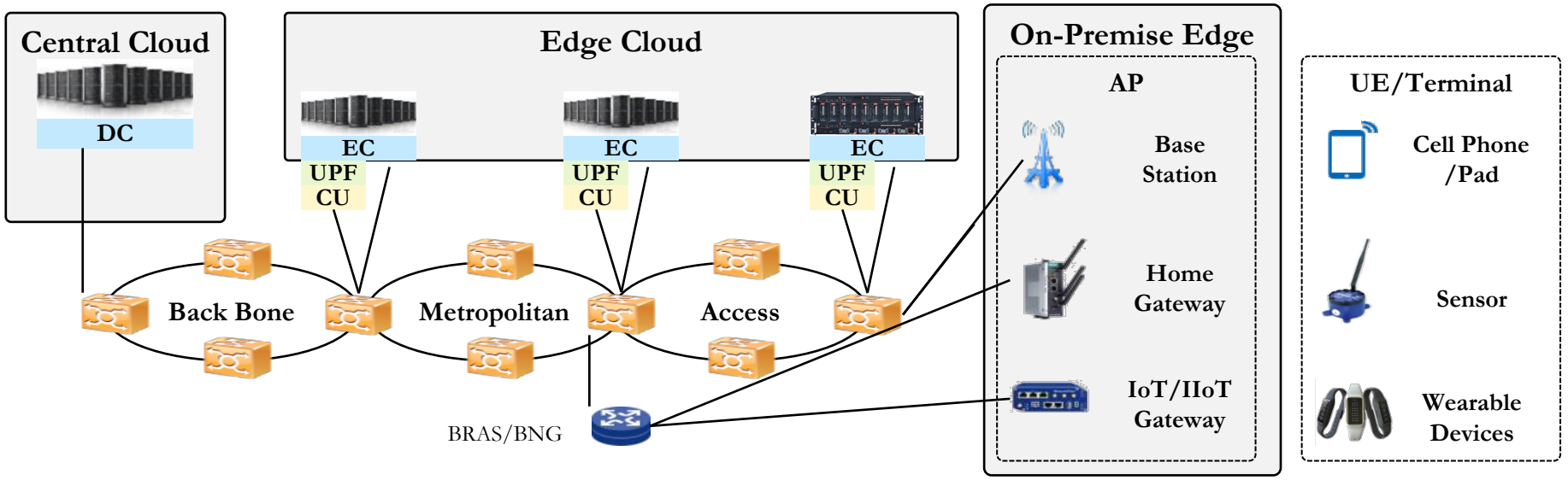
Aims at computing and network resource optimization by steering traffic to appropriate computing resources considering not only routing metric but also computing resource metric and service affiliation.

Note: Computing resource is a general term, it may include not only computing resource but also other resources like storage, etc.

ITU-T: CNC aims at computing and network resource joint optimization based on the awareness, control and management over network and computing resources.

CNC focus on the vision, scenarios, requirements, architecture and network function enhancements for future mobile core network and the telecom fixed, mobile, satellite converged network, but not for internet or routing area.

Rapid Development of integrated ICT Infrastructure



Facts in China Mobile

- CDN nodes in every city (**330+**) and major county (**250+**), with **25000+** servers installed
  - *These nodes can be upgrade to vCDN and then edge computing infrastructure*
  - *More diverse computing resource need to be provided;*
- More edge computing nodes will be setup in an on-demand manner
  - County aggregation **6000+,** Access aggregation **10,000+,** On-site **100,000+**

Increasing SP are offering the integrated computing and networking infrastructure.

Why dose the infrastructure develop so fast?

- **Users want** the best user experience such as low latency and high reliability, etc. .
- **Users want** the stable services' experience when moving among different areas.

How to meet the users' requirements?

- **Provide functional equivalency**
  - Deploy instances for the same service across edge sites for better availability
- **Keep the load balance for both static and dynamic scenarios**
  - by both upper layer and network layer solutions
- **Steer traffic dynamically to the "Best" Service Instance**
  - Traffic is delivered to optimal edge sites according to more status,e.g,computing (defining what 'best' is for each service)

However, the fact is ...

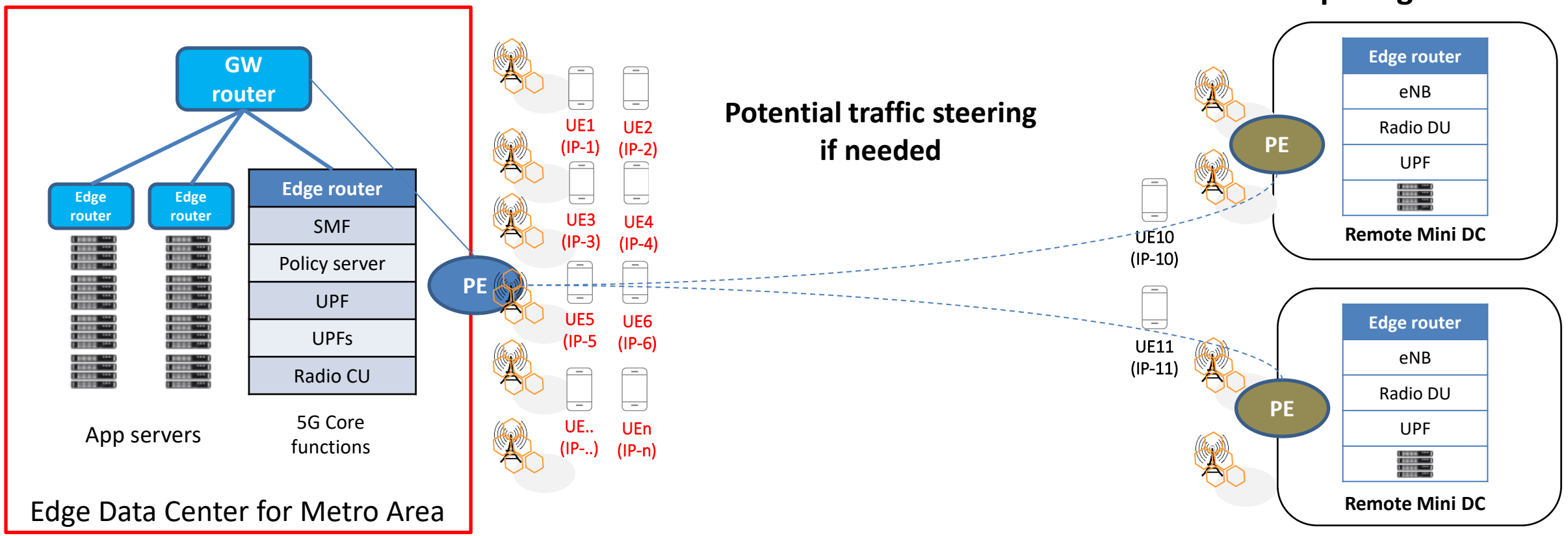
Edge computing has the advantage of ‘closest’, but in some cases,
the ‘closest’ is not the ‘best’.

Some reasons

- Closest site may not have enough resource, the load may dynamically change.
- Closest site may not have related resource, heterogeneous hardware in different sites.

High computing resources allocated at Metro Edge DCs (for large numbers of UEs at working time)

- More UEs in Metro Area
- High computing resource

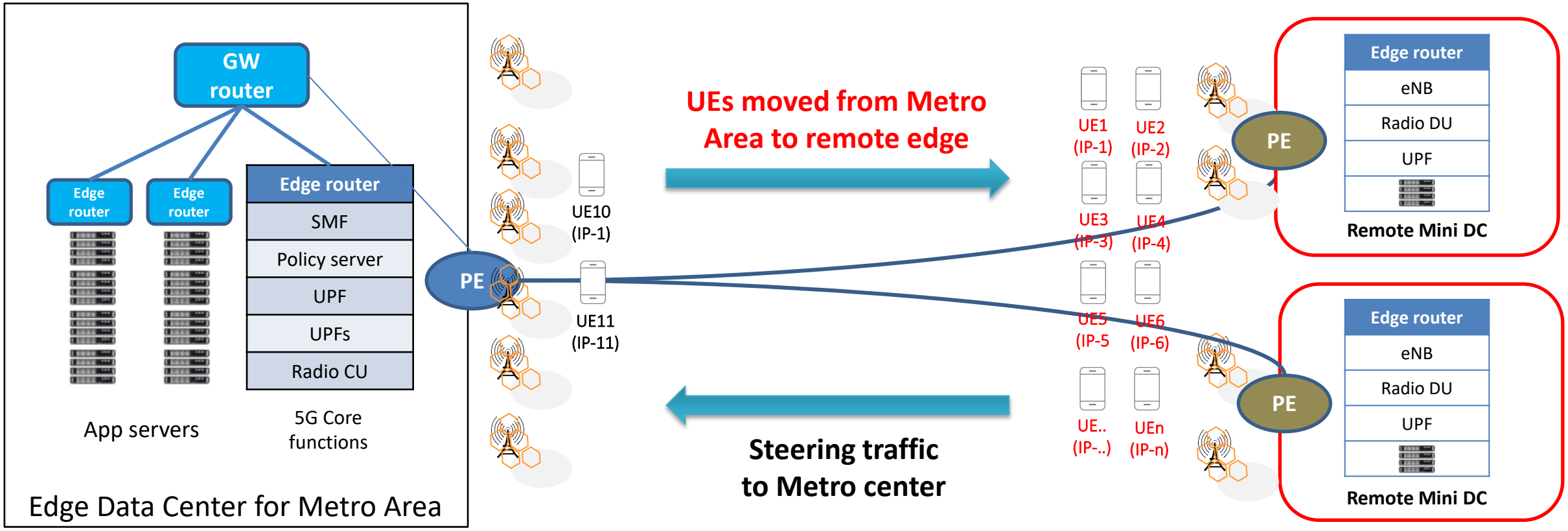


- Less UE closes to remote edge
- Limited computing resource

Weekend events at a remote site require high computing usage (only for 1~2 days, can't justify adding servers to the remote site)

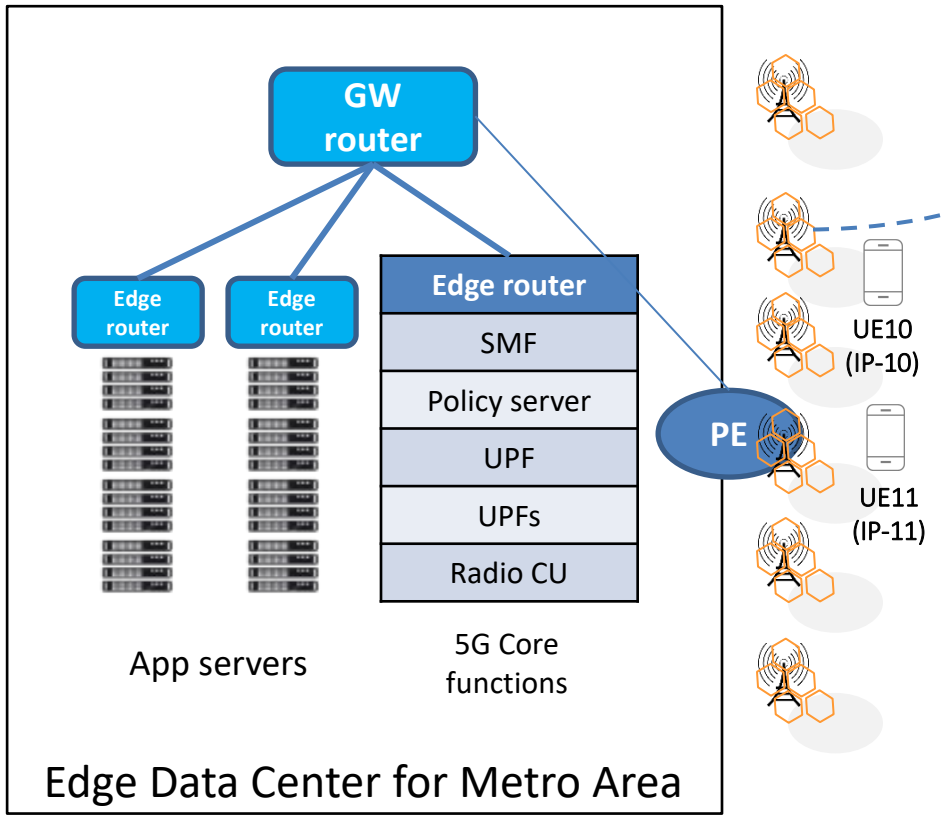
- Less UEs in Metro Area
- High computing resource

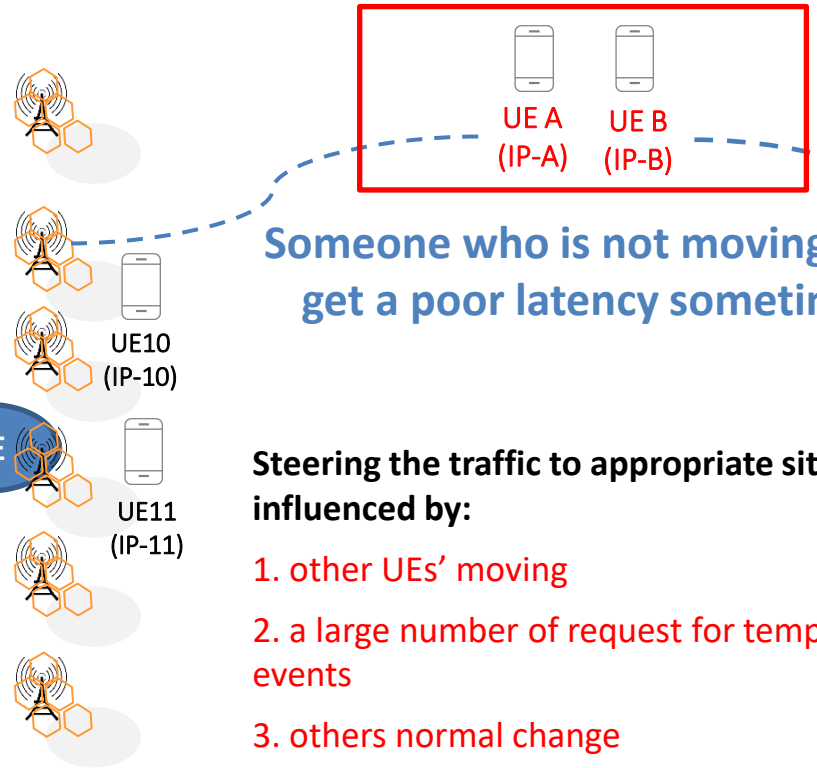
- More UEs closes to remote edge
- Limited computing resource



More optimal to steer traffic from SOME UEs to other Edge DC

- Less UEs in Metro Area
- High computing resource



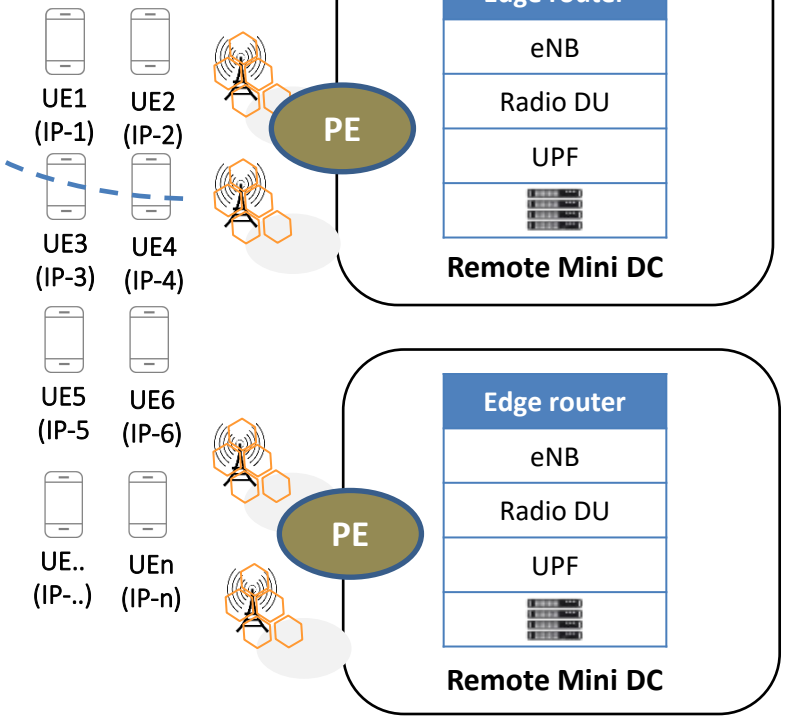


Someone who is not moving but get a poor latency sometime

Steering the traffic to appropriate sites influenced by:

1. other UEs' moving
2. a large number of request for temporary events
3. others normal change

- More UE closes to remote edge
- Limited computing resource



Considerations

High computing resources needed by UEs at a remote site for short period of time, which is not long enough to justify adding more computing resources at the remote site.

Traffic needs to be steered among different edge sites.

More thoughts

When steering traffic, what factors should be considered?

Some apps require both low latency and high computing resource usage or specific computing HW capabilities (such as local GPU); hence joint optimization of network and computing resource may be needed to guarantee the QoE.

Typical Application – AR/VR

Upper bound latency for motion-to-photon(MTP): includes frame rendering and requires less than **20 ms** to **avoid motion sickness,** consisted of**:**

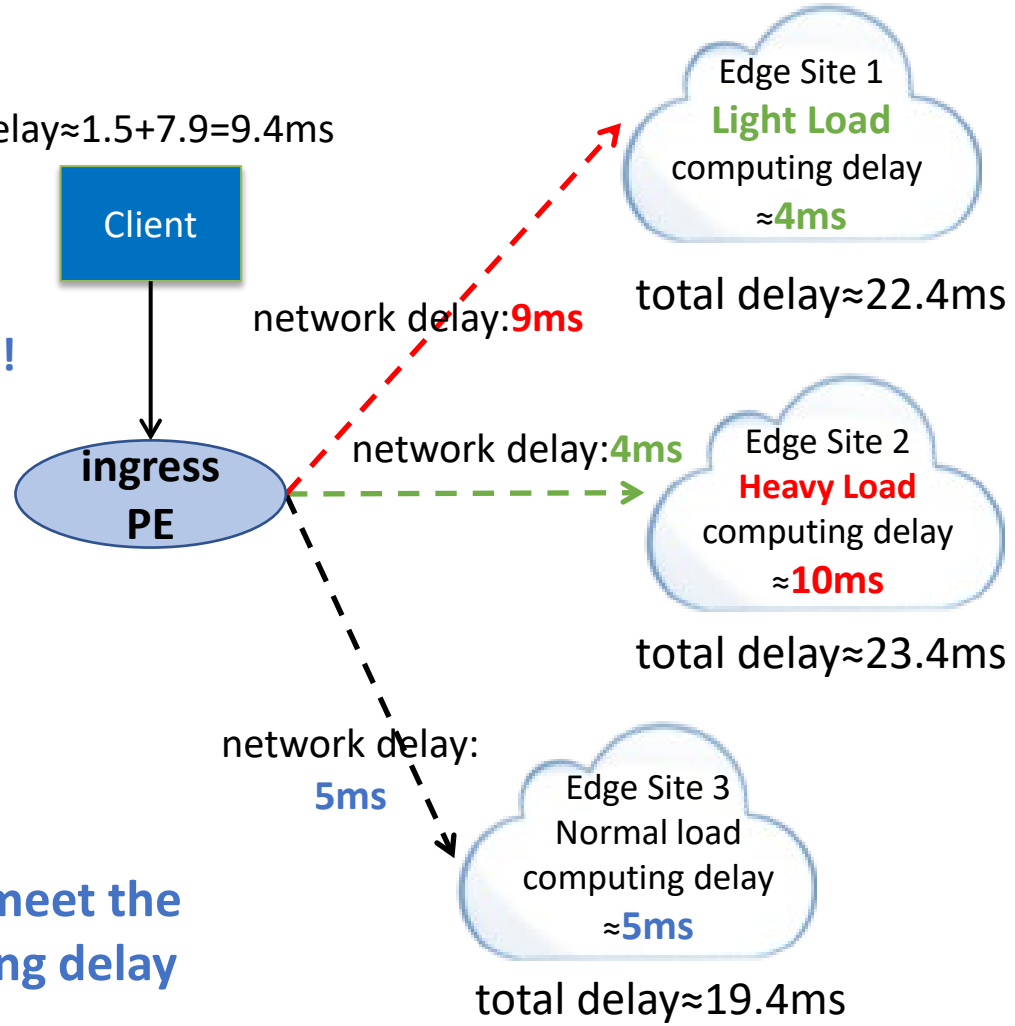
1. sensor sampling delay: <1.5ms (client)
2. display refresh delay: ≈7.9 ms(client)
3. frame rendering computing delay with **GPU≈ 5.5ms** (server)
4. network delay(budget) =20-1.5-7.9-5.5 = **5.1ms**(network)

Budgets for computing delay and network delay are almost equivalent!!

- choose edge site 1 according to load only, total delay≈22.4ms
- choose edge site 2 according to network only, total delay≈23.4ms
- choose edge site 3 according to both, **total delay≈19.4ms**

It can't meet the total delay requirements or find the best choice by either optimize the network or computing resource:

Require to dynamically steer traffic to the appropriate edge to meet the E2E delay requirements considering both network and computing delay

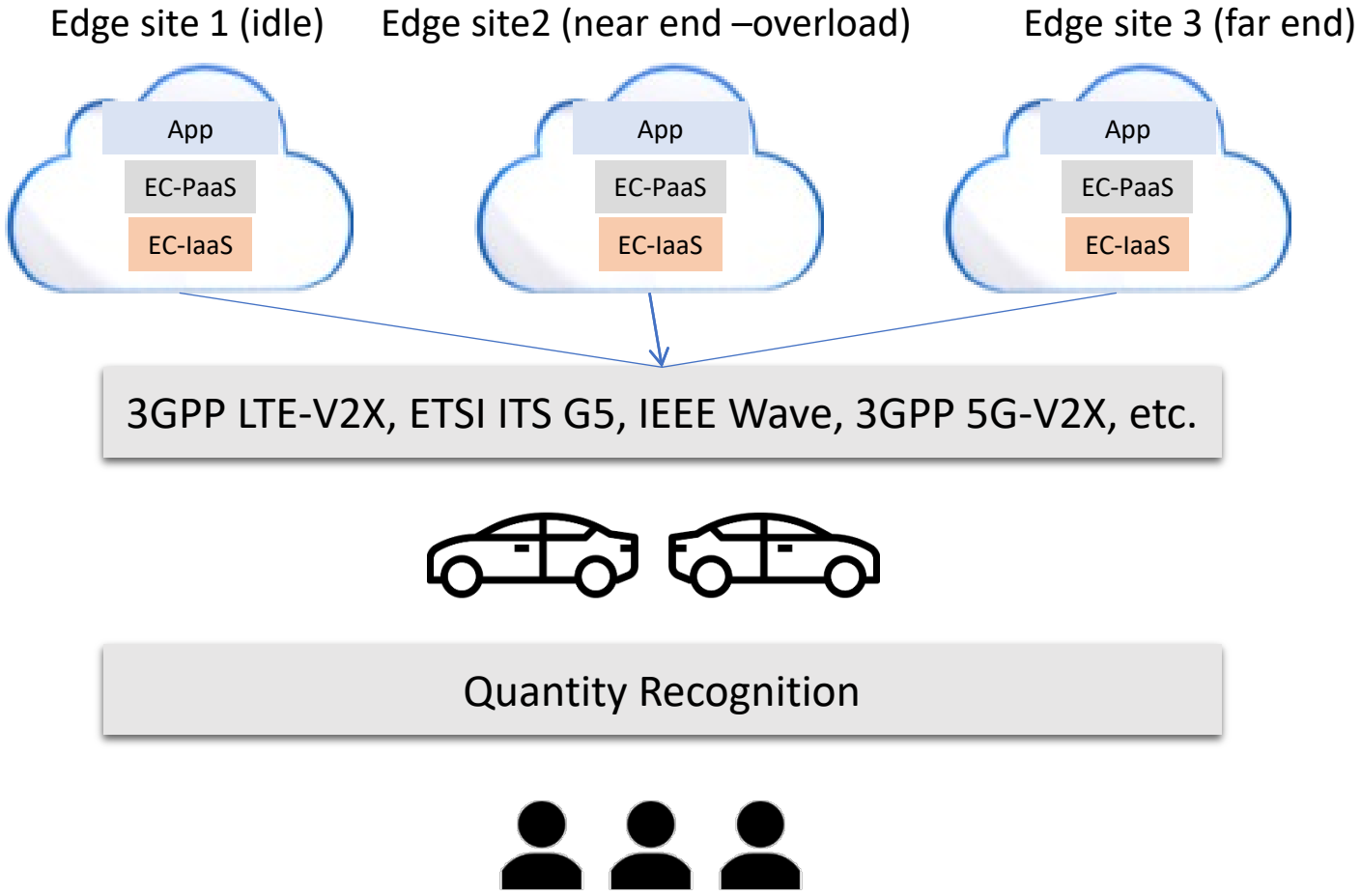


PS: Computing resources have a big difference in different edges, and the 'closest site' may be good for latency but lacks GPU support and should therefore not be chosen.

More Applications - Intelligent transportation

Connected Car

Function	Requirement
Driving-assist	Low Latency
HD and HP Map	High bandwidth



Video recognition at intersection

Function	Requirement
Safety Monitoring	Low Latency
Data analysis	High bandwidth

The load of network and edge sites may change dynamically and rapidly!!

Considerations

Those apps require both low latency and high/specific computing resources have the almost equivalent budgets for computing delay and network delay, and the load of network and edge sites may change dynamically and rapidly.

When steering traffic, the real-time network and computing resource status should be considered at the same time in an effective way.

Preliminary Conclusions

- Traffic needs to be steered among different edge sites.
- When steering traffic, the real-time network and computing resource status should be considered at the same time in an effective way.

Thank you!