Towards an SDN-based Mobile Core Networks (MCN)

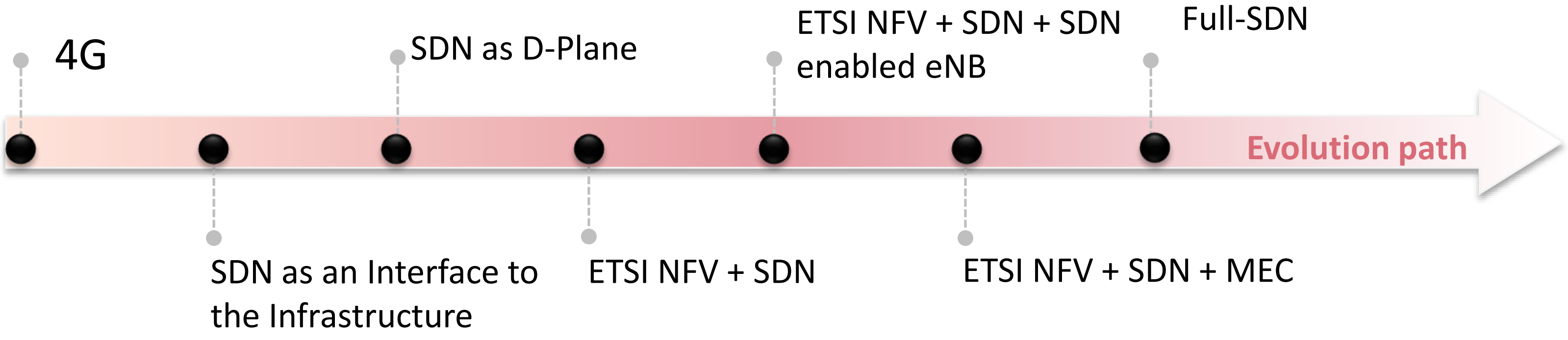
Xueli An and Artur Hecker
Huawei Technologies, European Research Center Munich
Xueli.An@huawei.com and Artur.Hecker@huawei.com

BUILDING A BETTER CONNECTED WORLD

VDE/ITG-Fachgruppe 5.2.4, “5G System Architecture”, Dec. 10. 2015

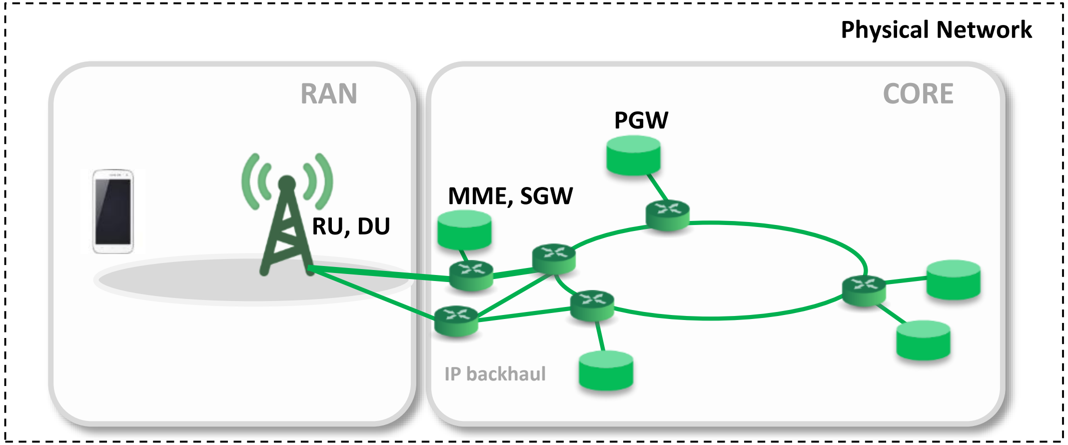
Contents

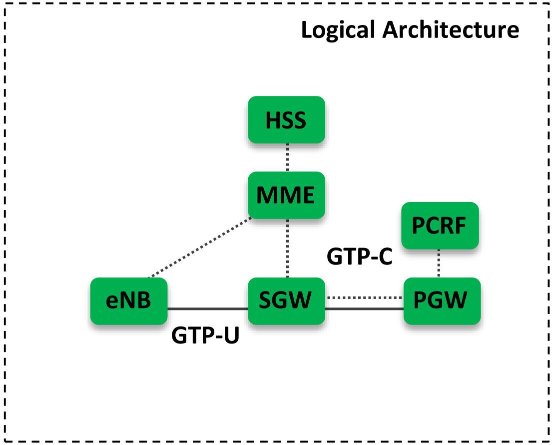
MCN Evolution Path



From Theory to Practice: 5G on Testbed

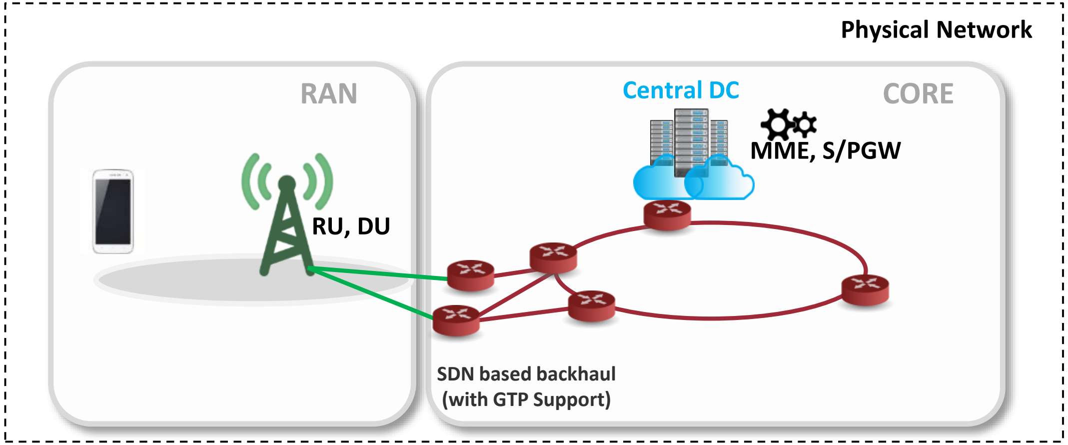
Current Generation (4G): LTE SAE / EPC

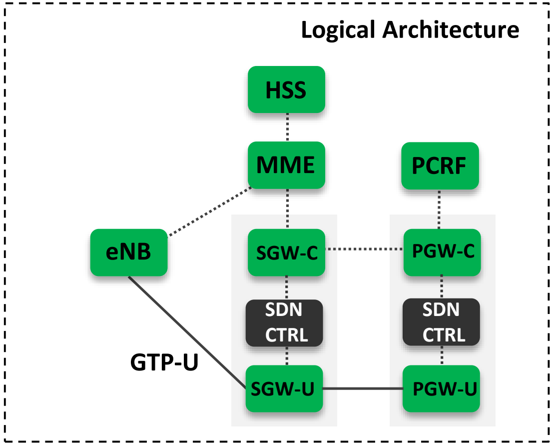




- System Architecture:
  - Pre-configuration: fixed infrastructure, fixed architecture, manual setup
  - Single architecture: one size fits all
  - LTE entities becoming bottlenecks (MME in C-plane, PGW in D-plane)
- Non-native transport: overlaying in both D-Plane and C-Plane
- Non-optimized service provisioning: packets are touched by many entities, user states are spread in multiple network elements
- Cost: All dedicated hardware, high CAPEX and high OPEX

Evolution (1) SDN as an Interface to the Infrastructure

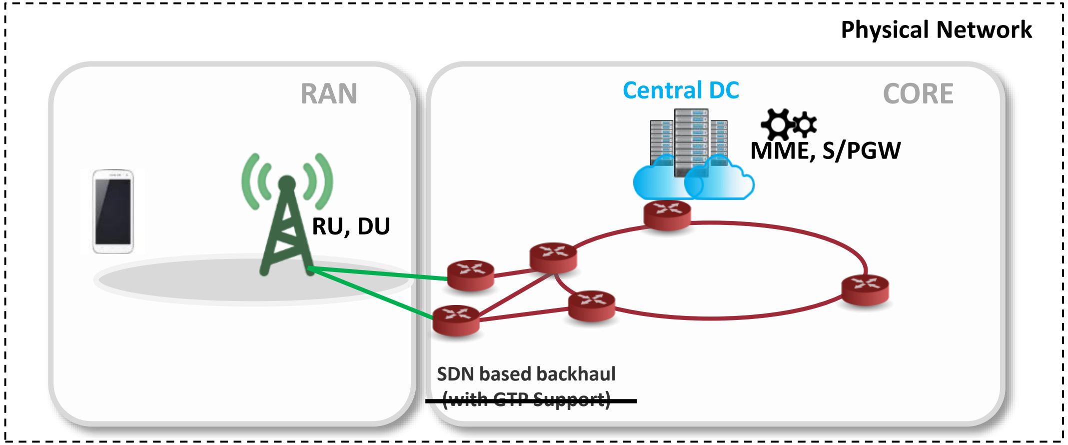


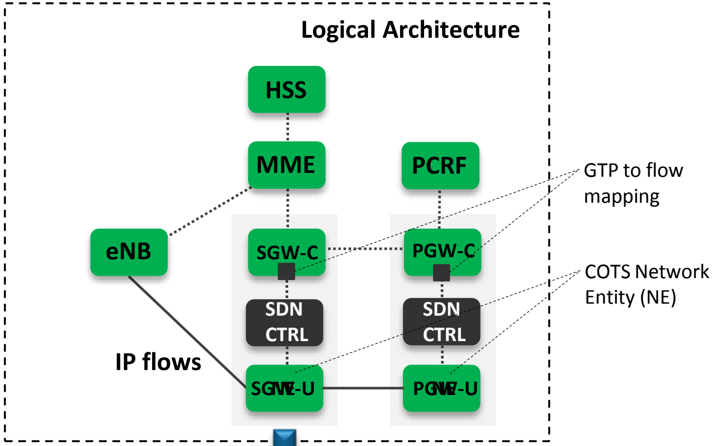


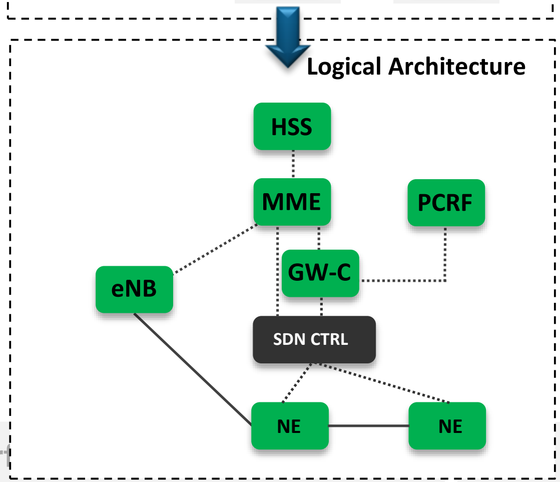
- Virtualized network function (VNF) approach
  - Core network function like MME, S/PGW-C are implemented as software and run on Commercial-Off-The-Shelf (COTS) hardware.
- Decoupled the MCN C- and D-plane.
  - More dynamic flow distribution over the infrastructure

- SDN's advantages are not fully leveraged.
  - SDN only sets flow paths for GTP-U data bearers
- Logical architecture is untouched. Same as conventional LTE.
  - Directly map from "physical box" to "virtual box"
  - Pre-configuration
  - Single architecture: one size fits all

Evolution (2) SDN as D-Plane: Get rid of GTP Tunnels



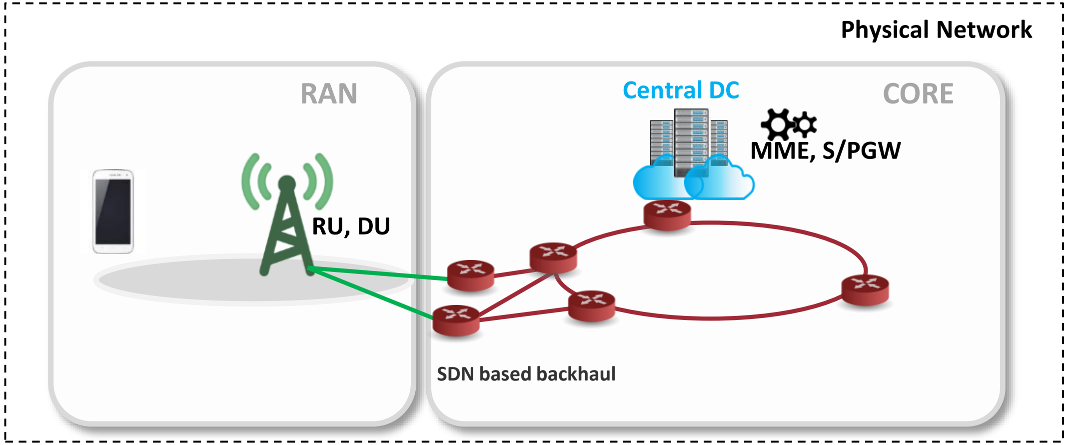


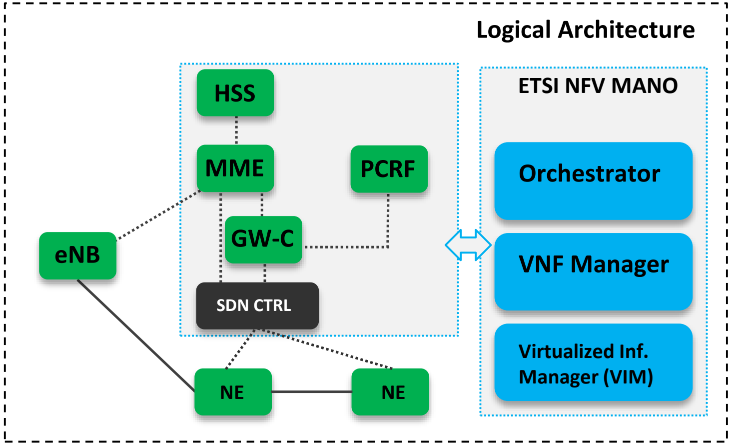


- GTP-U is replaced by flow entries directly
- Native data plane transport: less overhead on the data plane

- The C-plane of the logical architecture is barely touched. Similar to conventional LTE.

Evolution (3): ETSI NFV + SDN

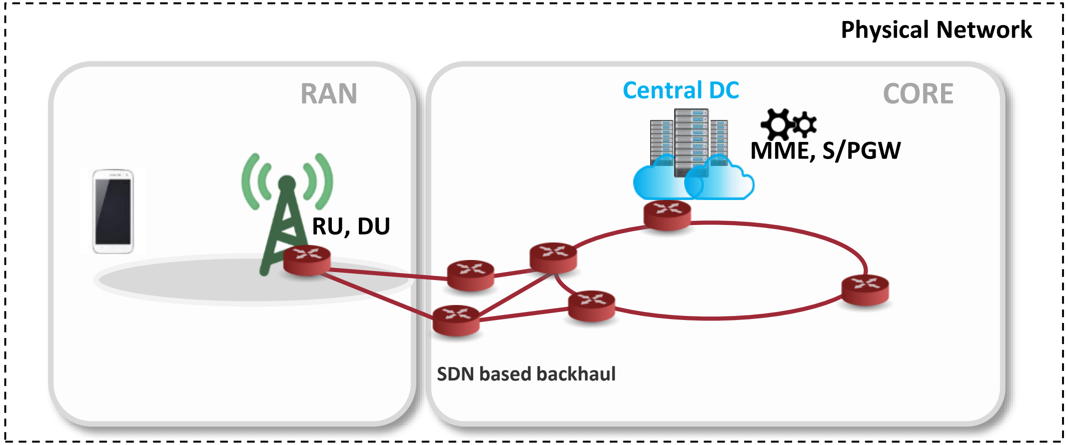


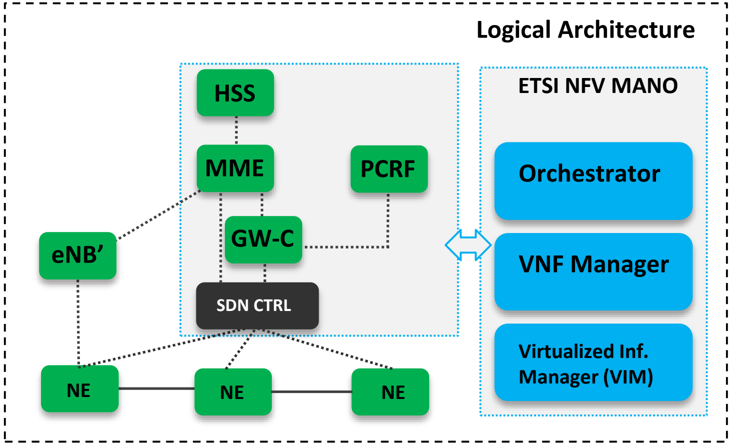


- Bring system agility
- Respond on-demand to the dynamic needs of the traffic and services running over it.
- Allows overcoming bottlenecks more easily

- The C-plane of the logical architecture is barely touched. Similar to conventional LTE.
- Extra management effort

Evolution (4): ETSI NFV + SDN + SDN enabled eNB

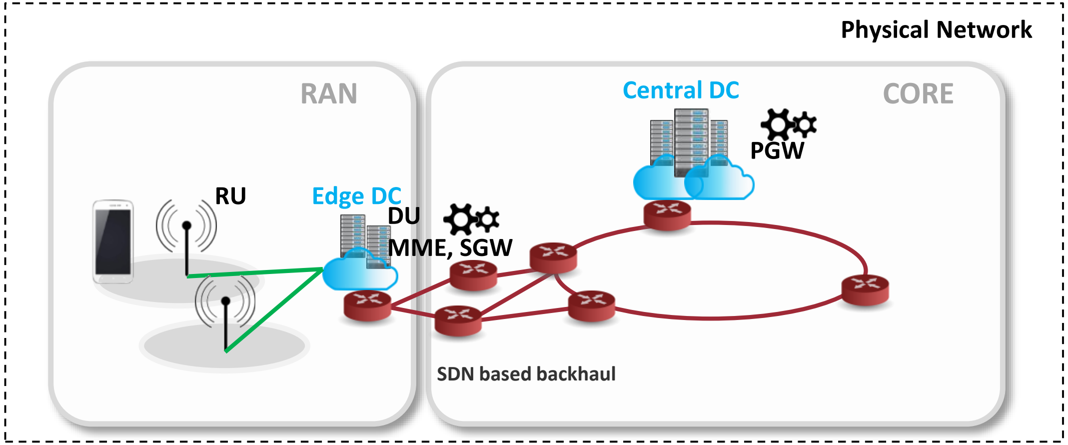


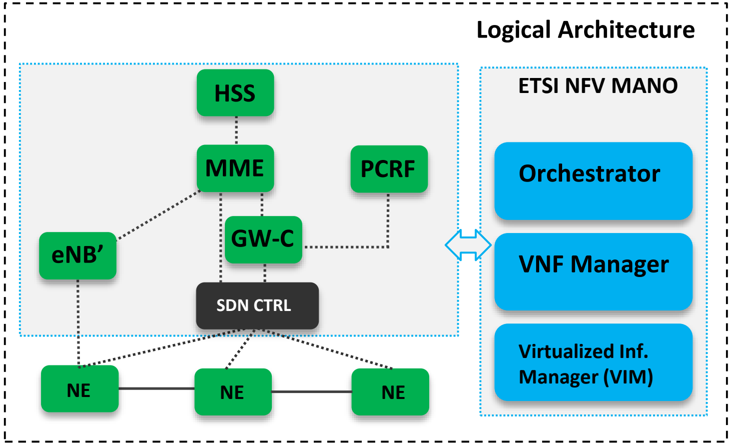


- A new functional split on eNB -> eNB' + NE
  - eNB' contains radio part and C-plane related functions
  - Flow-related parts is controlled by SDN control apps.
- Native traffic distribution in the infrastructure, depending on the SDN capabilities

- The C-plane of the logical architecture is barely touched. Similar to conventional LTE.
- Extra management effort

Evolution (5): ETSI NFV + SDN + MEC

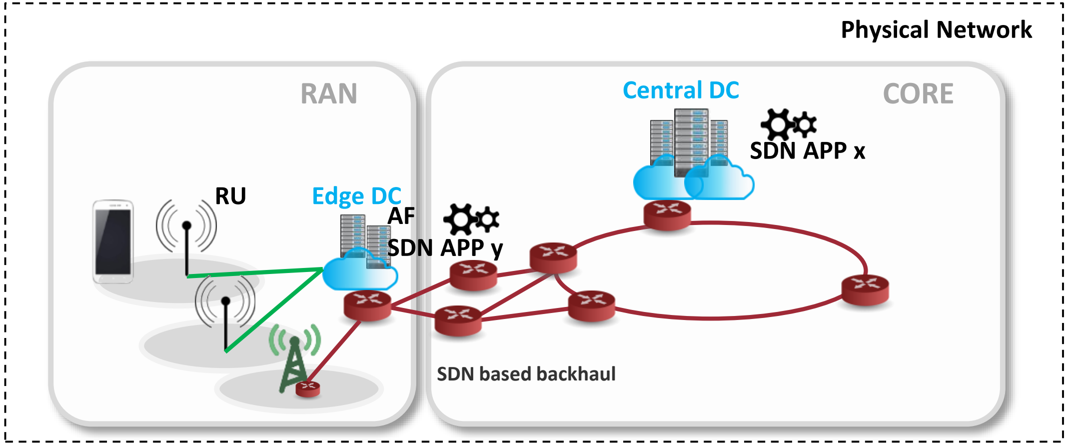


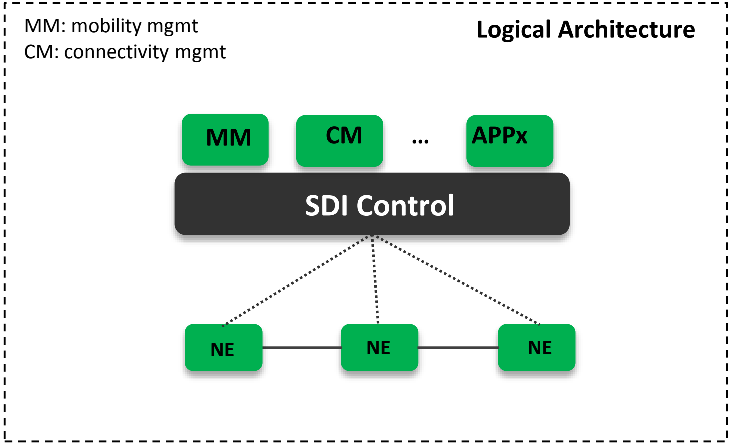


- Resource pooling at the edge.
- Enable C-RAN model.
- Enable core network functions to be placed at the edge -> reduced C-plane latency -> Concept merging between the "Core" and the "Edge"

- The C-plane of the logical architecture is barely touched. Similar to conventional LTE.
- Extra management effort

Evolution (6): Programmable infrastructure - Full-SDN



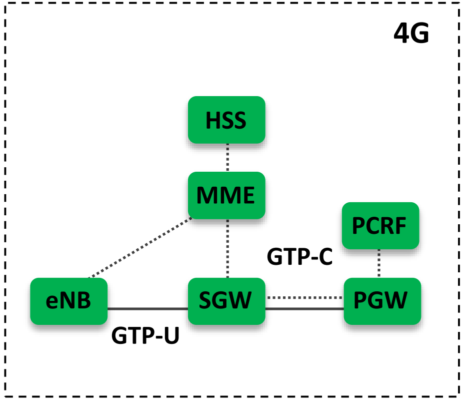


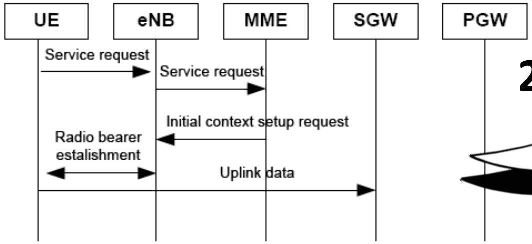
No fixed architectures anymore - architecture is a programme.

- A programmable functional split
- One unique control plane
- MCN with full programmability: no bottlenecks, flexibility, agility, etc.

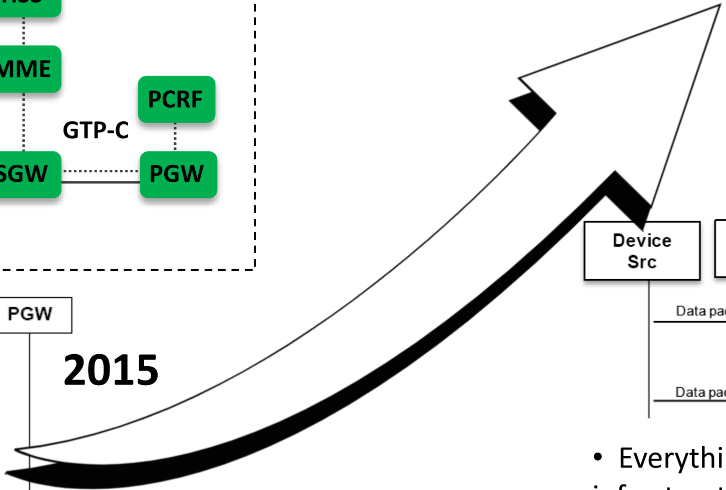
- Distributed system limits, potentially too complex to realize?
- Performance-wise: carrier-grade?

EPC vs. Full SDN MCN

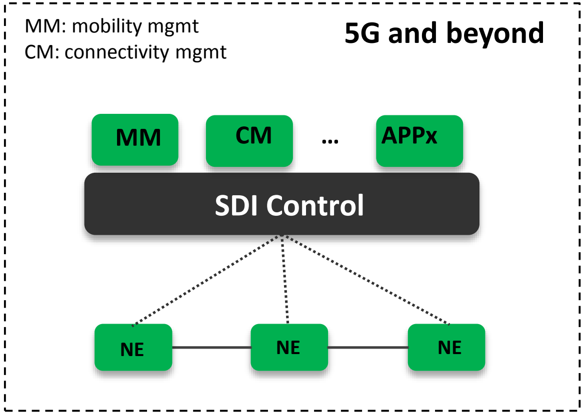


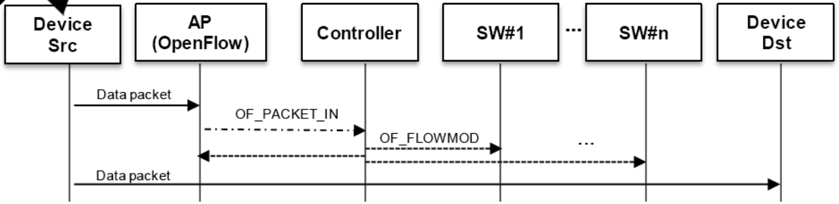


2015



2020+

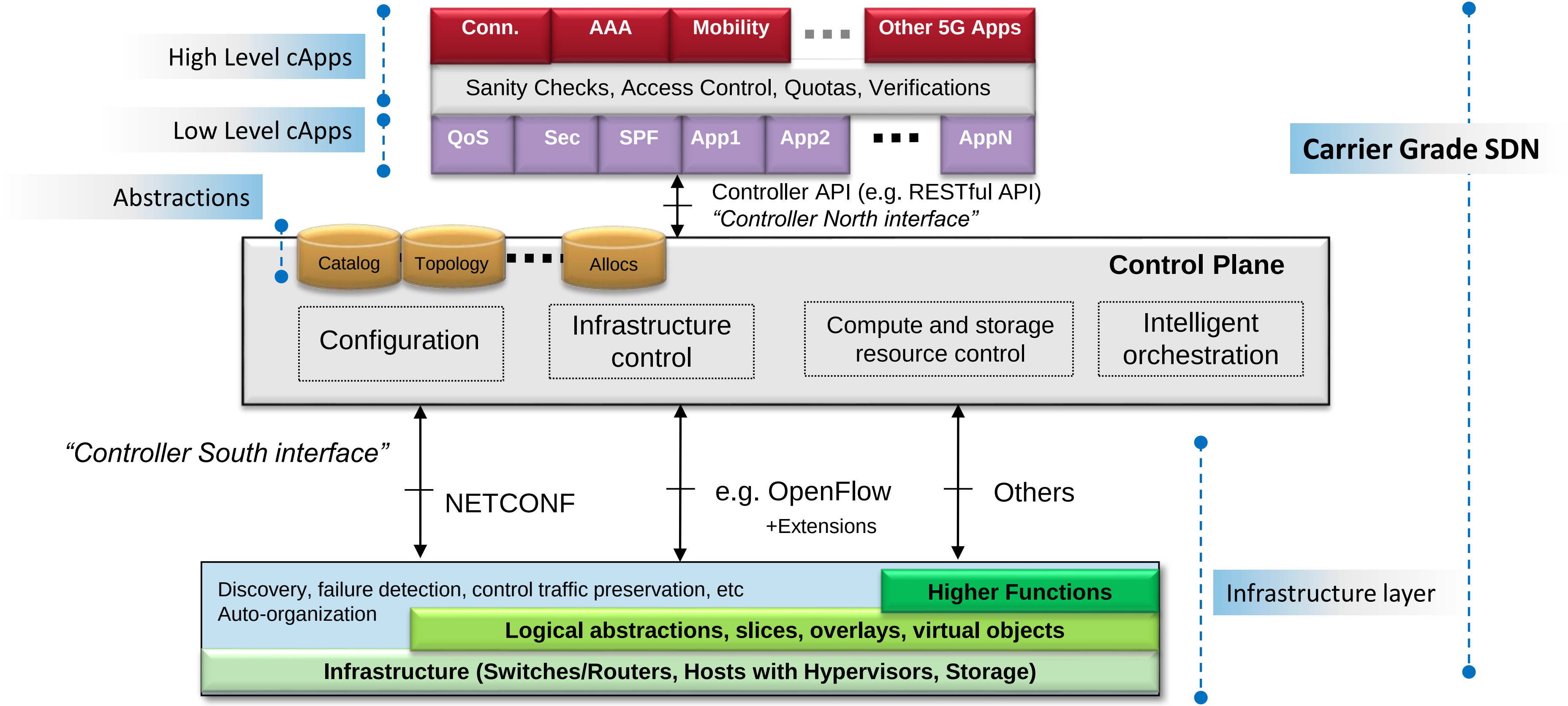




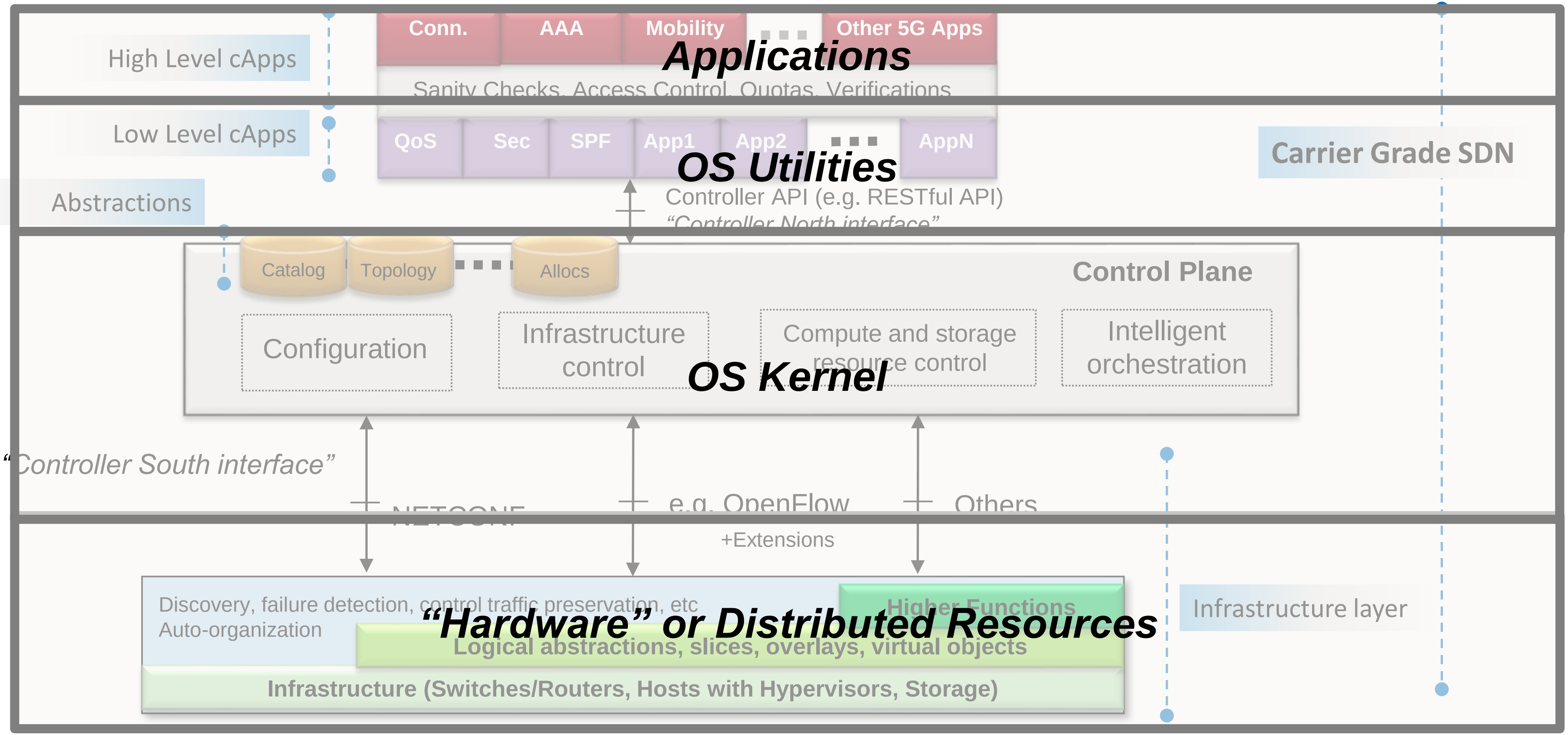
- Everything is event-based; events are dispatched from the infrastructure elements to cApps.
- CP runs on the same infrastructure (in-resource ctrl). cApps dynamically decide what to do.

- Pre-provisioned exchanges in both C- and D-Planes, realized as overlays

MCN as Full-SDN

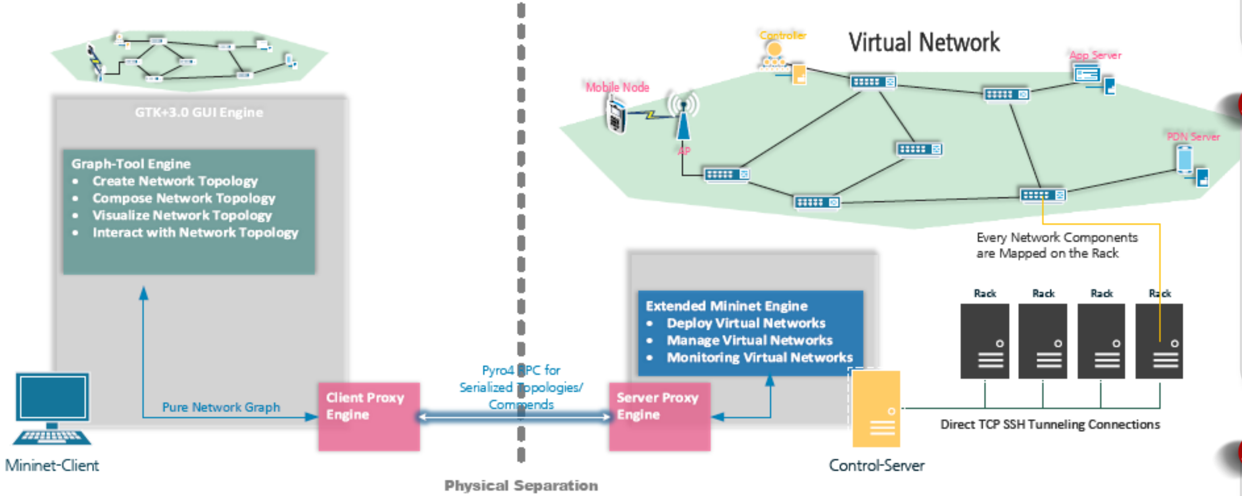


MCN as Full-SDN



From Theory to Practice: 5G on Testbed

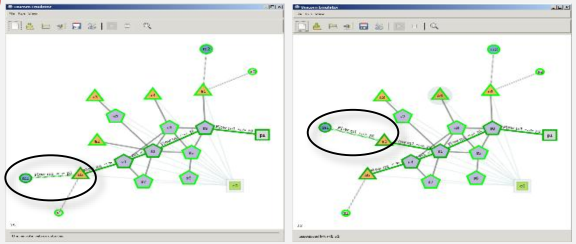
Product of
Future Network Technologies
ERC in Munich, Huawei Technologies



Client Specifications:
- OS Platform: Linux Ubuntu
- GUI Engine: GTK+3.0
- Client Proxy Engine: Pyro4 RPC

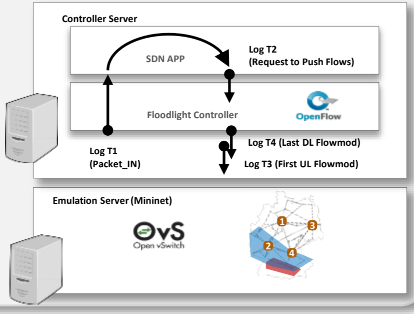
Server Specifications:
- OS Platform: Linux Ubuntu
- Mininet: Customized Mininet 2.2.1
- Server Proxy Engine: Pyro4 RPC

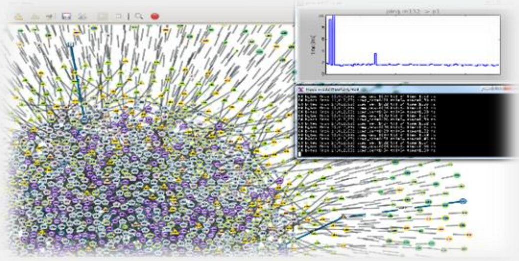
We pursue a radical goal of tuneless, anchor-less and NFV-less mobility support in SDN



MCN support: service request, mobility, paging, etc.

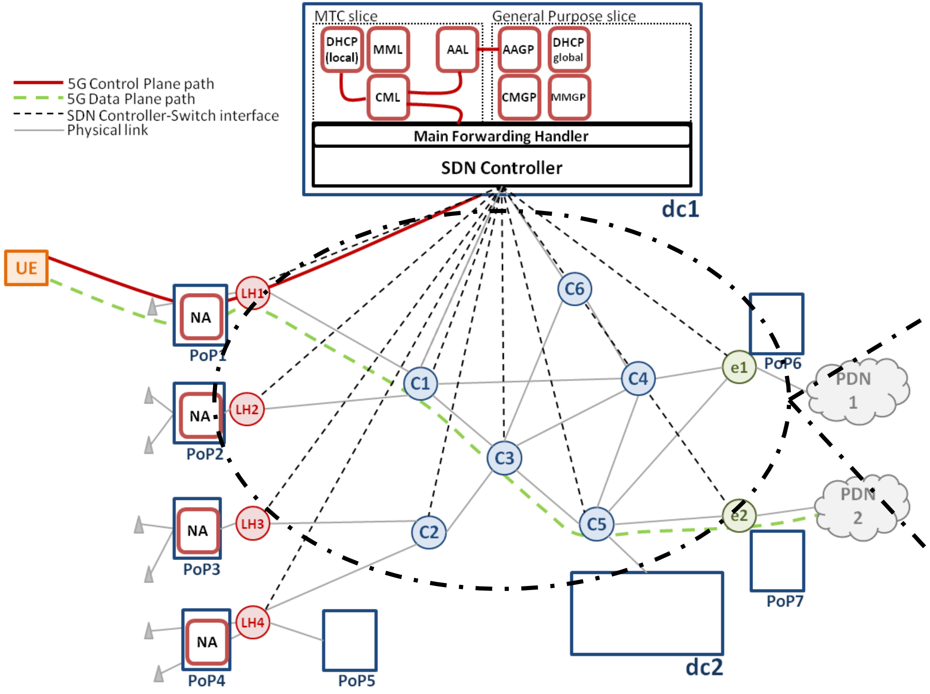
It could measure!

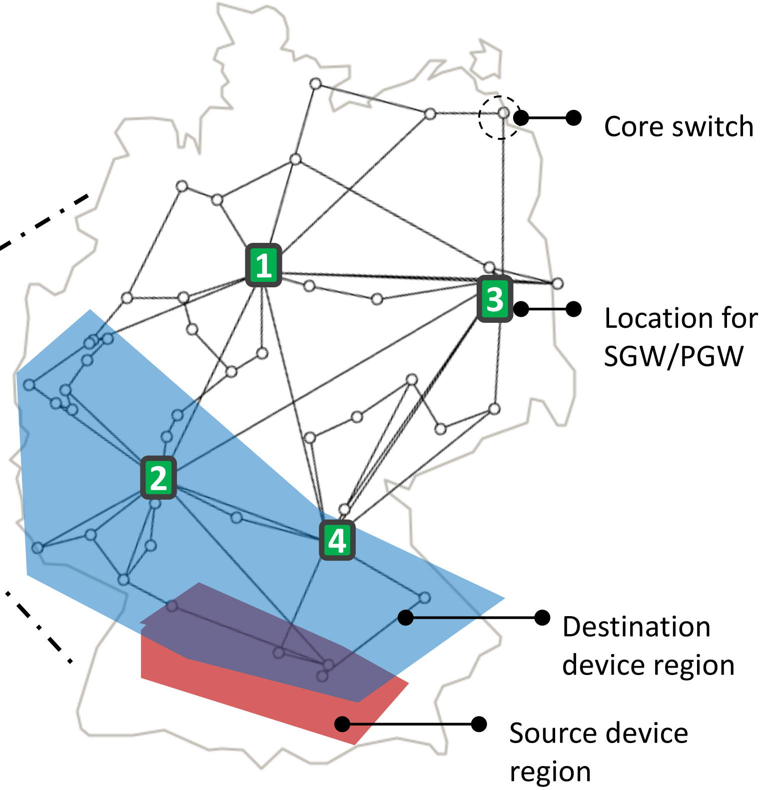




It also scales!

From Theory to Practice: 5G on Testbed





C-Plane/U-Plane functions:
AAL, AAGP: Authentication Authorization Local, AA General Purpose
CML, CMGP: Connectivity Management Local, CM General Purpose
MML, MMGP: Mobility Management Local, MM General Purpose
AN: Access Node
GW: Gateway

Forwarding elements:
LH Last Hop Forwarding Element
c Core Forwarding Element
e Network Entry Point

Final Word

- General programmability = major vulnerability
  - **The very feature bears the main risk**
  - *What will happen when thousands of programmable entities will be waiting in a large-scale environment for close-to-realtime programmatic control by other virtual entities?*
- Openness => Error-proneness
  - How to efficiently support access control for multi-tenancy scenario?
- How to pursue 6-9 reliability and “towards 0 latency”?

JOIN US IN

BUILDING A BETTER CONNECTED WORLD

THANK YOU

HUAWEI

EUROPEAN
RESEARCH
CENTER