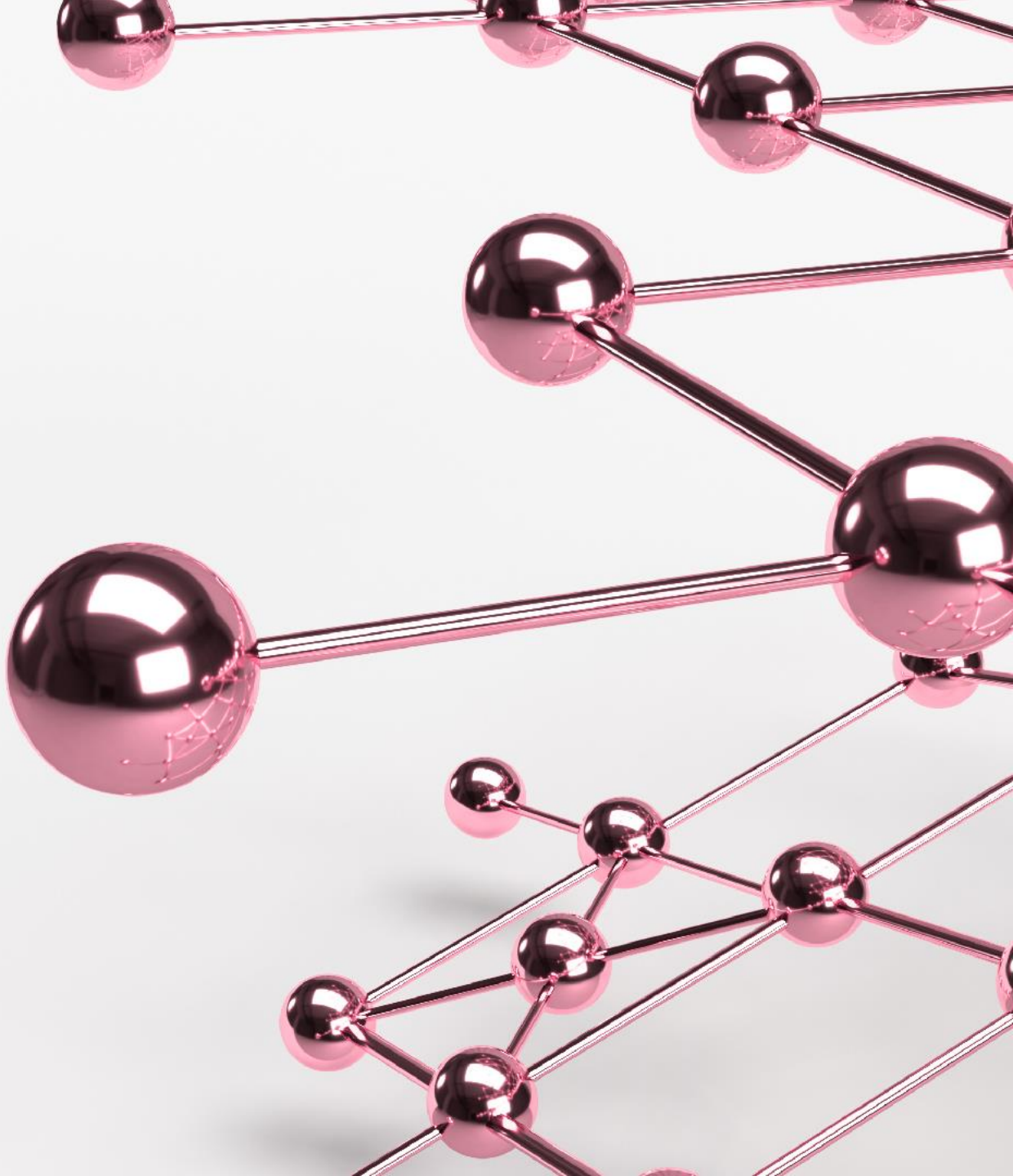
DESIRE6G

PROGRAMMABILITY AND DISTRIBUTED INTELLIGENCE: A WAY TOWARDS PREDICTABILITY?



Co-funded by the European Union

6GSNS

06/10/2023

WHY DESIRE6G?

What is the difference between D6G and the other 6G projects?

We study

- How end-to-end network programmability helps in solving really challenging use cases / KPIs (such as below ms latency)
- How to solve the complexity problem of centralized control and optimization with a distributed agent-based system
- And how can we put this together as simply as possible with other innovative methods, like AI-driven telemetry, blockchain-based federation and a DLT-backed software security framework
- So D6G has a **bottom-up** view and focuses on proof of concept **demos** to validate the value proposition


In the Networked Society
anything which benefi
from being c
be c
So besides further improving radio characteristics, we need to consider architectural changes too
the main promises for 5G – besides performance?
A converged infrastructure (radio, transport and compute)
Automation and programmability throughout the system
Integrated Machine Intelligence throughout the system
Features for supporting highly demanding and diverse performance throughout the system

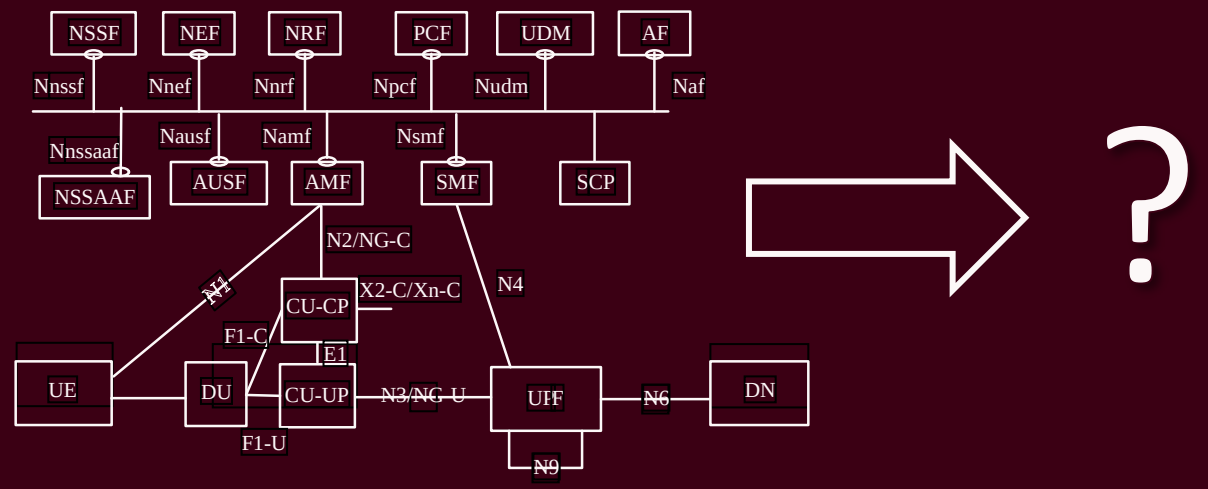
CHALLENGES FOR THE 6G / 2030 ARCHITECTURE

Main questions of all architecture discussions:

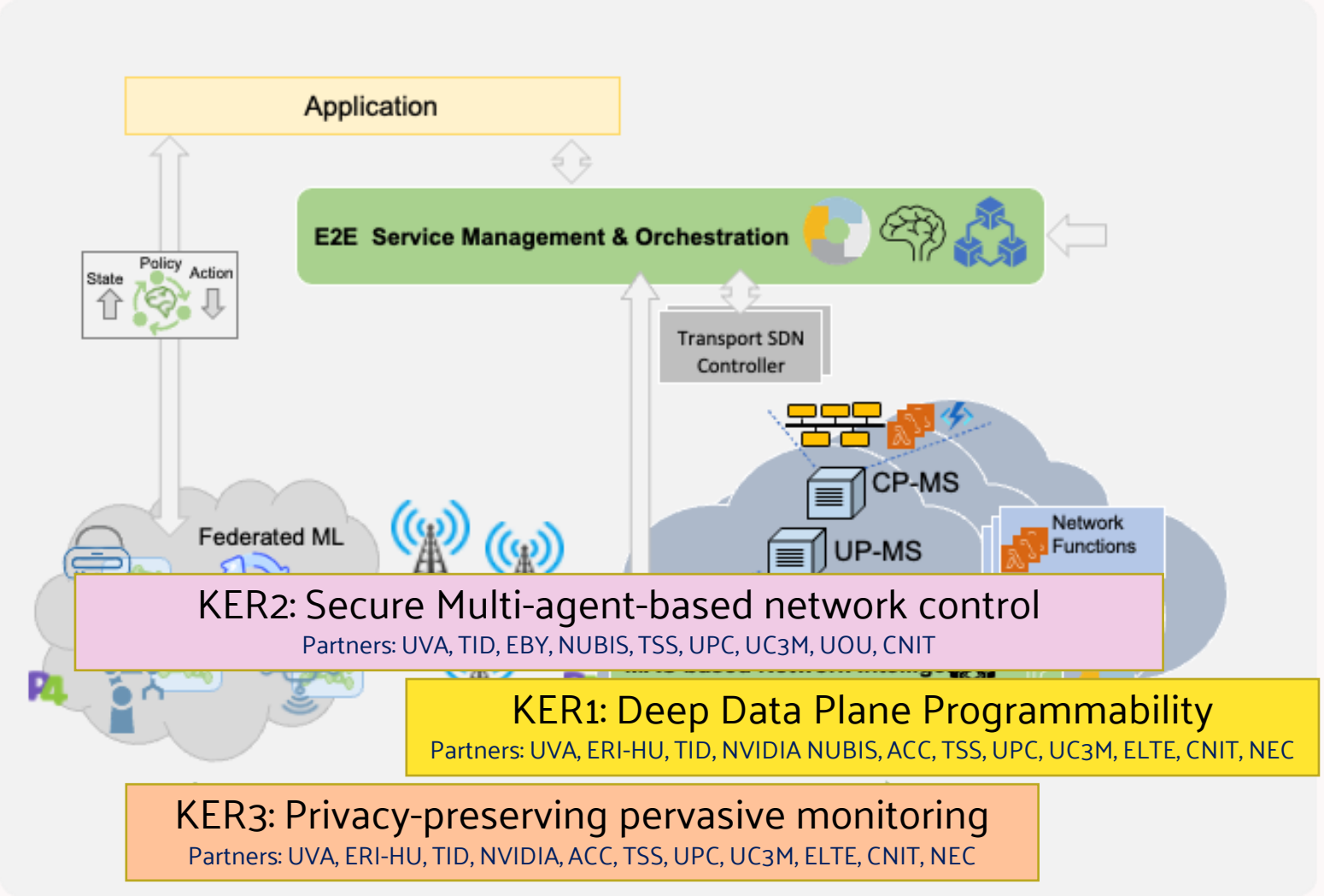
- How should the functions be grouped / split?
- How should the interfaces and procedures look like?

5G was addressing complexity issues, but only with partial success:

- “Service Based Architecture” (SBA) became heavier and less cloud-native than expected
- User plane remained mainly node-based, no “cloud-native” evolution happened there
- Too detailed standards, less room for vendor innovation
- The standard does not really count on using IT frameworks/tools to simplify the architecture



D6G ARCHITECTURE



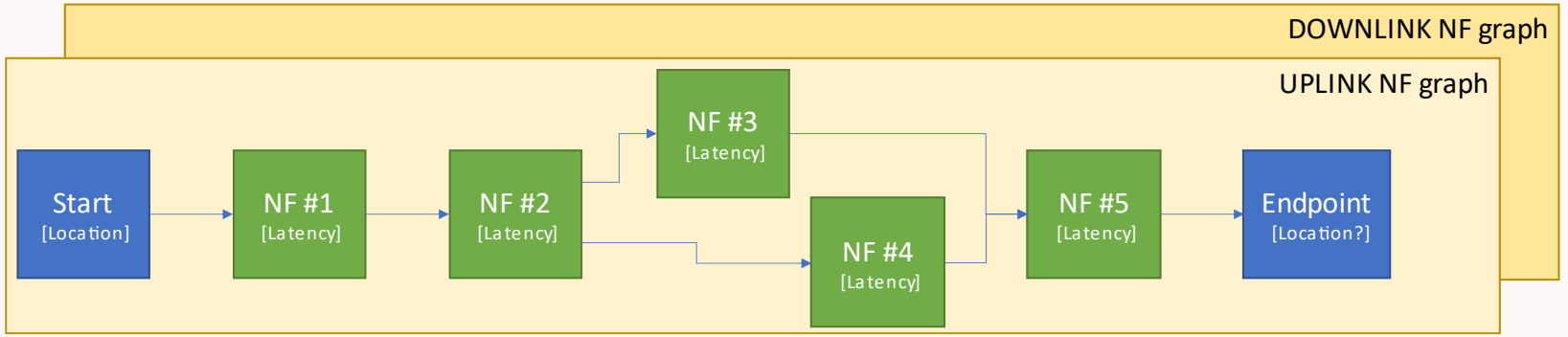
Innovation

- Intelligent Applications
- End-to-end and per-domain intent-based orchestration
- Lightweight blockchain-based federation
- xURLLC services
- Serverless Architecture
- RAN-Core Convergence
- Cloud-to-edge continuum
- Secure Distributed Intelligence
- E2E Data Plane Programmability
- Multi HW acceleration
- E2E Telemetry

BASICS OF DP PROGRAMMABILITY: SERVICE GRAPH

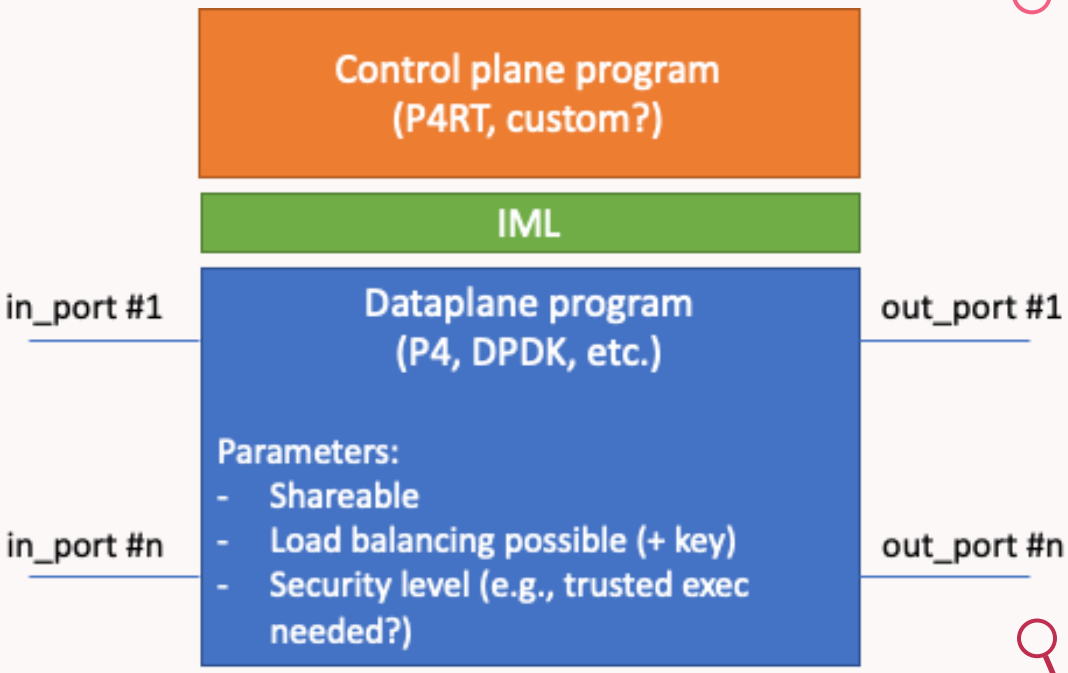
NF (or service) graph

- Describes the DP of one service, e.g., Internet access, robot control @MEC
- Shareable NFs of the service template can be used by all the UEs of the same slice
- One graph per direction (UL / DL) – the functionality is not always the same



BASICS OF DP PROGRAMMABILITY: NF

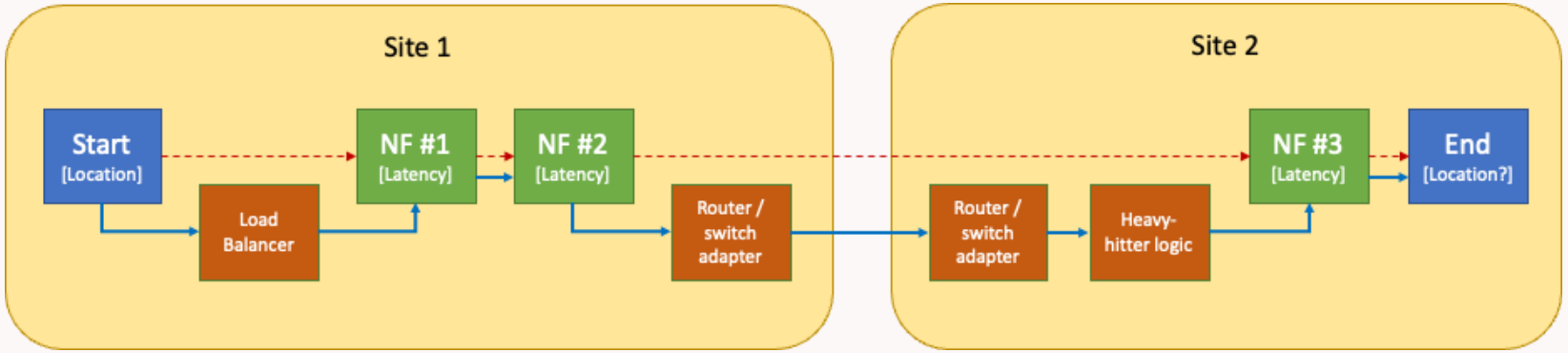
- Multiple in and out (virtual) ports
  - NF chain can have branches
  - A pipeline is a high-level execution graph
- DP program
  - Can use in_port as input parameter (e.g., for classification)
  - Must select an out_port as a result of the DP program
  - Might buffer the packet (non P4)
  - Might generate a packet (non P4)
- IML
  - NF DP (also hardware) abstraction layer: CP sees unified view
  - Implements infrastructure functions (e.g., adapters, load balancing, HH, slicing)
- CP program
  - Works on the abstract (IML) view of the DP
  - Can manage multiple DP flavours (e.g., CPU, NPU, ASIC)



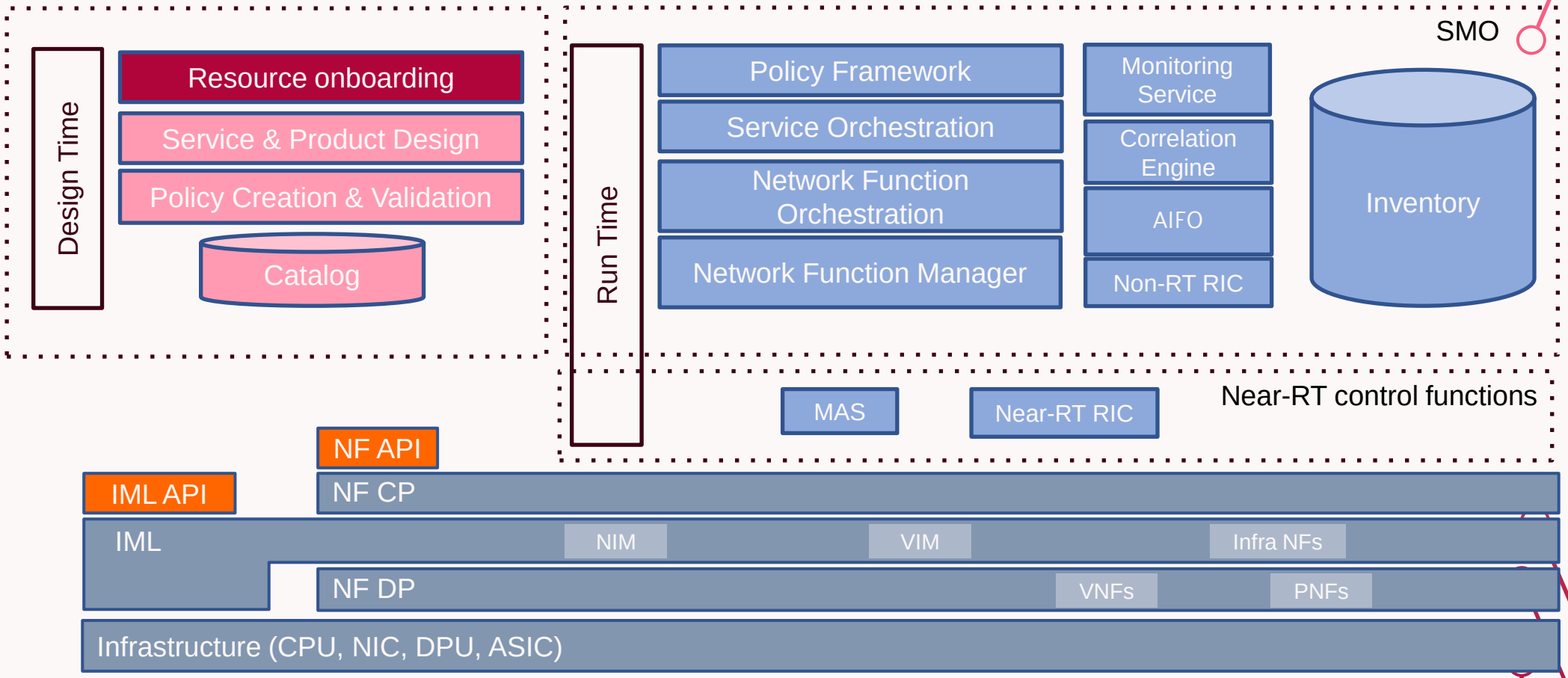
IML AND SERVICE DEPLOYMENT

- During service deployment IML can (sometimes must) add further NFs to the graph
  - Transport adapters: adapt to transport between two sites
    - Probably via non-PDP entities – we need to connect these domains
  - Network slicing (both separation and QoS)
  - Load balancing for a given NF or graph fragment
  - Heavy-hitter pattern (kind of load balancing)

} Transparent optimizations for NFs



SERVICE MGMT & ORCHESTRATION



REAL-TIME SERVICE CONTROL

- Multi-Agent System (MAS) is a distributed control function
- Closed-loop control
  - Telemetry collection
  - Predictions (possibly with AI)
  - Actuation / reconfiguration

  } MAS can insert telemetry NFs into any graph or use infrastructure based data collection points
- Long-time control: via SMO
  - E.g., new QoS slice is needed, resource blocks on closer sites are needed
- Short-time control: via IML and VIM
  - E.g., add more (local) resources, modify load balancing rules, simple error signals

DESIRE6G

THANKS!

Gergely Pongracz

Gergely.Pongracz@ericsson.com

Co-funded by the European Union

6GSNS

DESIRE6G has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096466.
Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the granting authority can be held responsible for them.