Teraflow SDN as control plane for Next Generation Central Office (NGCO) in hybrid TSN/optical environments

CTTC Parc Mediterrani de la Tecnologia - Castelldefels Barcelona, Spain

Pietro Piscione - Nextworks

Presentation outline

OCTAPUS project

- Motivation and main objectives
- Validation and Final demonstration
- Traffic scenarios
- Next Generation Central Office (NGCO) SDN-based Control plane

SDN Coordinator PoC with TFS

- Functional architecture
- Mapping to TFS components
- TFS software extensions
- Initial tests with Netopeer2 SDN Agent

OCTAPUS project – Brief overview

Optical Circuit switched Time sensitive network architecture for high-speed Passive optical networks and next generation Ultra-dynamic & reconfigurable central office environments

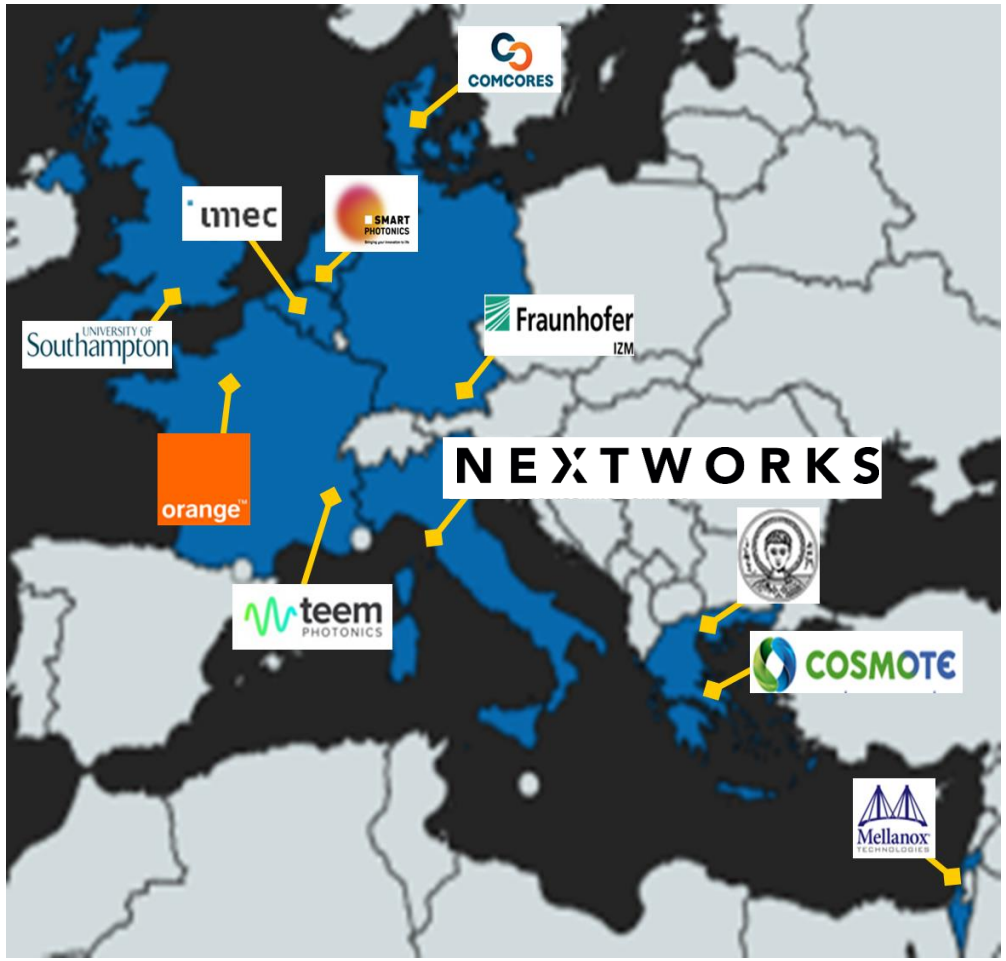
Duration: over 3.5 year - Started on September 2022

Motivation: Skyrocketing capacity demands are posing a new strict latency-oriented framework challenging 5G infrastructure and calling for new architectural changes at the key aggregation infrastructure being in local proximity to the subscribers: **the Central Offices (COs)**

Objective: develop an agile, low-cost, and energy-efficient technology framework for high-speed PON and **N**ext **G**eneration **C**entral **O**ffice (NGCO) environments

Web site: http://octapus-ict.eu/

The consortium



Funded by the European Union

Grant agreement No. 101070009

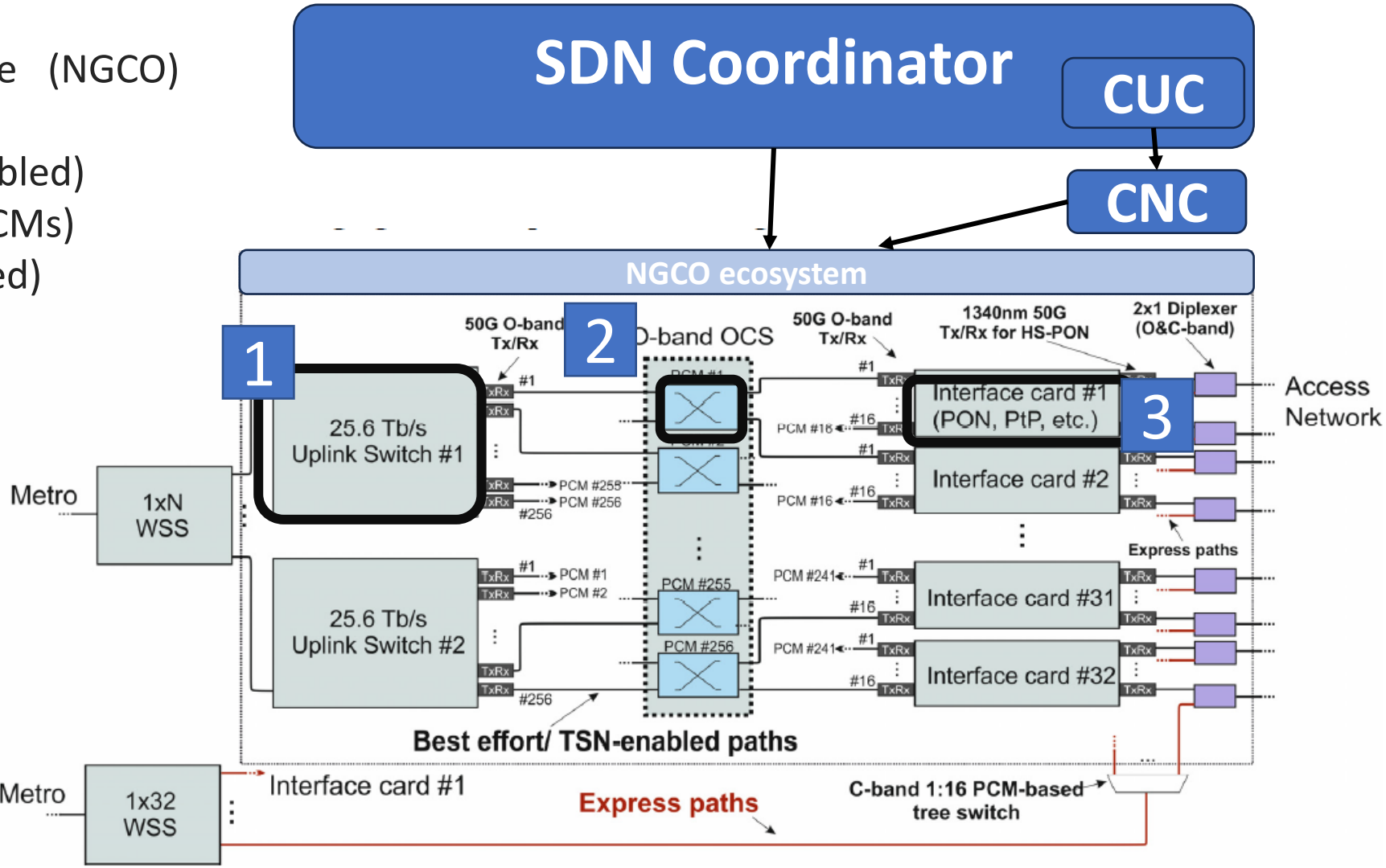
NGCO architecture

Next Generation Central Office (NGCO) composed of:

1. Two Uplink switches (TSN enabled)
2. One OCS (composed of 256 PCMs)
3. 32 interfaces card (TSN enabled)

TSN/optical SDN Coordinator for the end-to-end traffic management through the NGCO control plane

Centralized User Configuration (CUC) and Central Network Controller (**CNC**) for supporting TSN flows setup and management



Target scenario

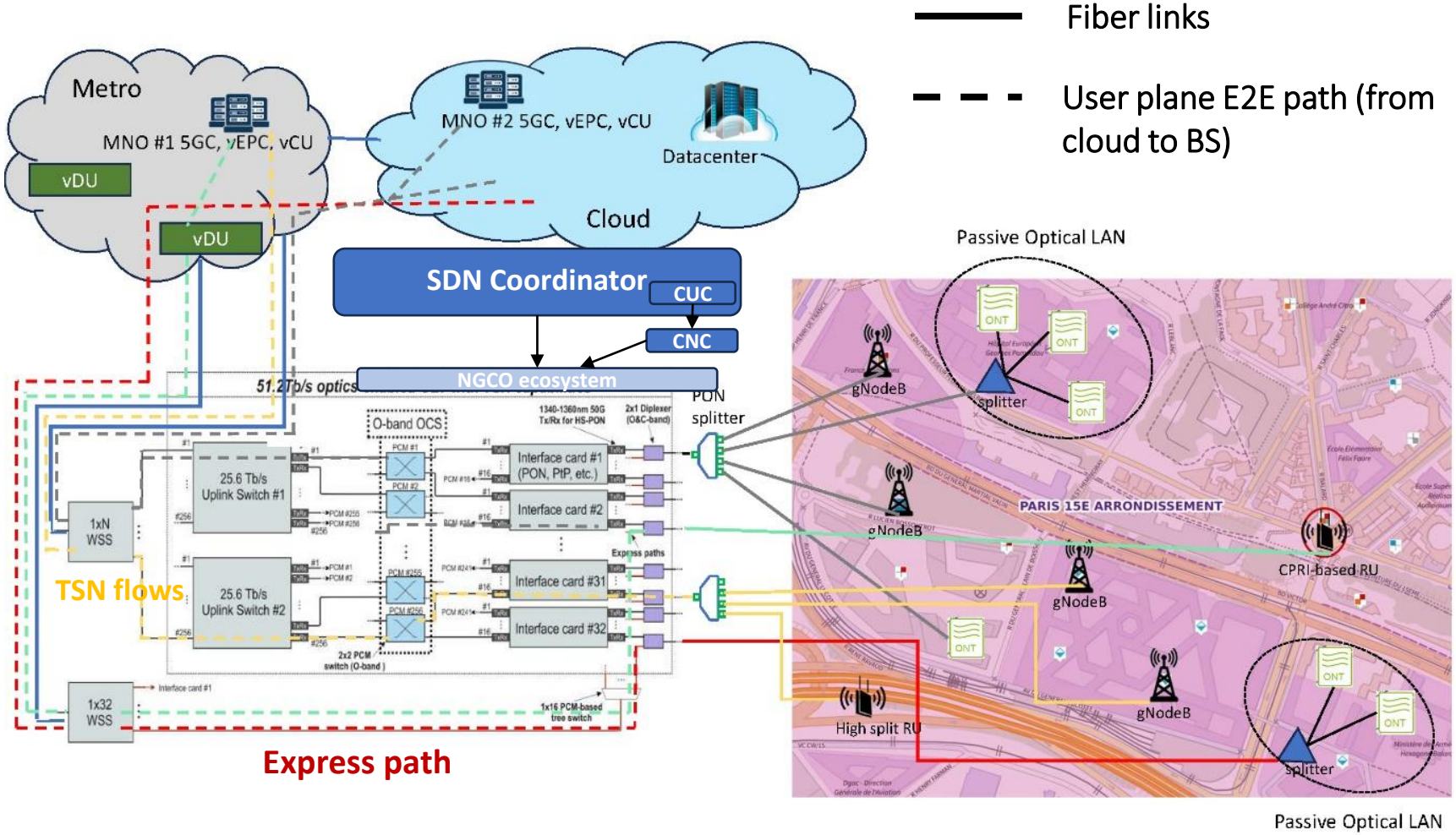
Dense area use case* in Paris arrondissement 15

Types of traffic envisioned in OCTAPUS:

- Best effort standard ethernet
- TSN flows
- Express

Target scenarios:

- TSN flows provisioning
- Uplink balancing

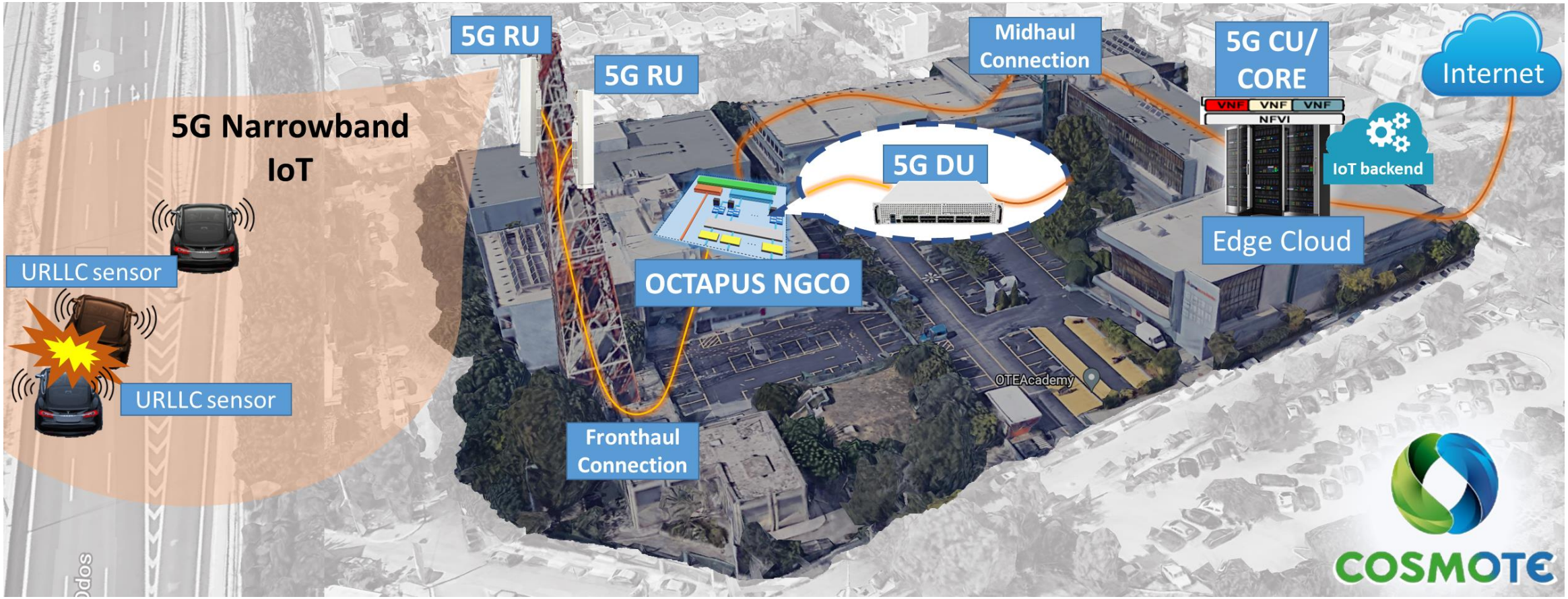


*Details on Use case available in D2.1 deliverable (it'll be released soon)

Experimental validation

Demonstrate a scalable **NGCO** architecture and validate its advanced optical component technologies through a series of lab and field trials in time-sensitive applications scenarios

- Lab trial of prototypes
- Final Field Demo @ Network Operator



SDN Coordinator - High-level architecture

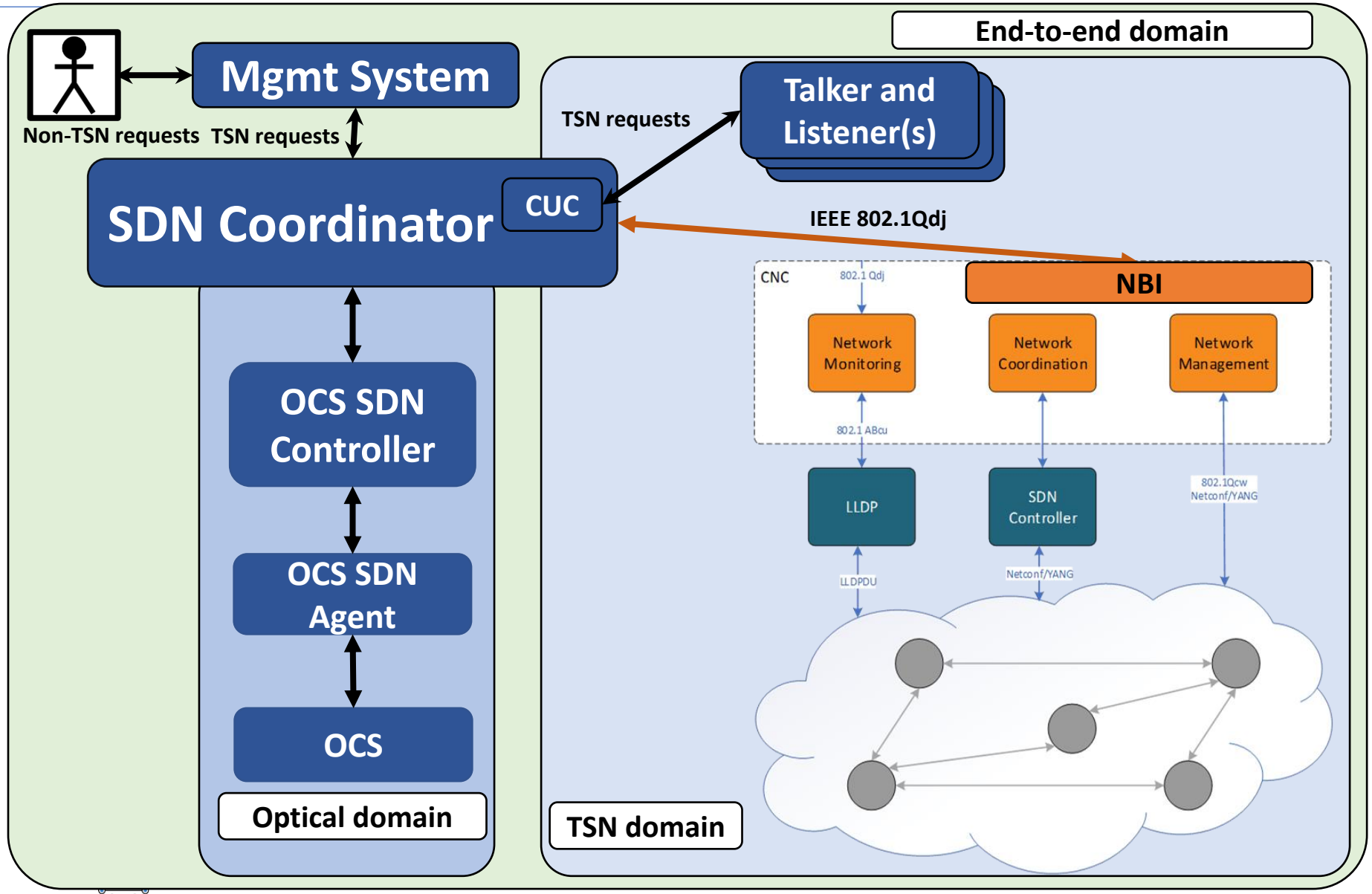
Traffic requirement requests from:

- Talker/Listener(s) (follow the **IEEE802.1Qdj** spec.)
- Network operator

CNC in charge of the control plane of **TSN domain**

SDN Coordinator:

- End-to-end domain view
- Interact with the CNC through the **CUC** for TSN control and information
- Optical (interacts with the OCS SDN Controller)



SDN Coordinator – Building blocks

SDN Coordinator would have some internal **building blocks**:

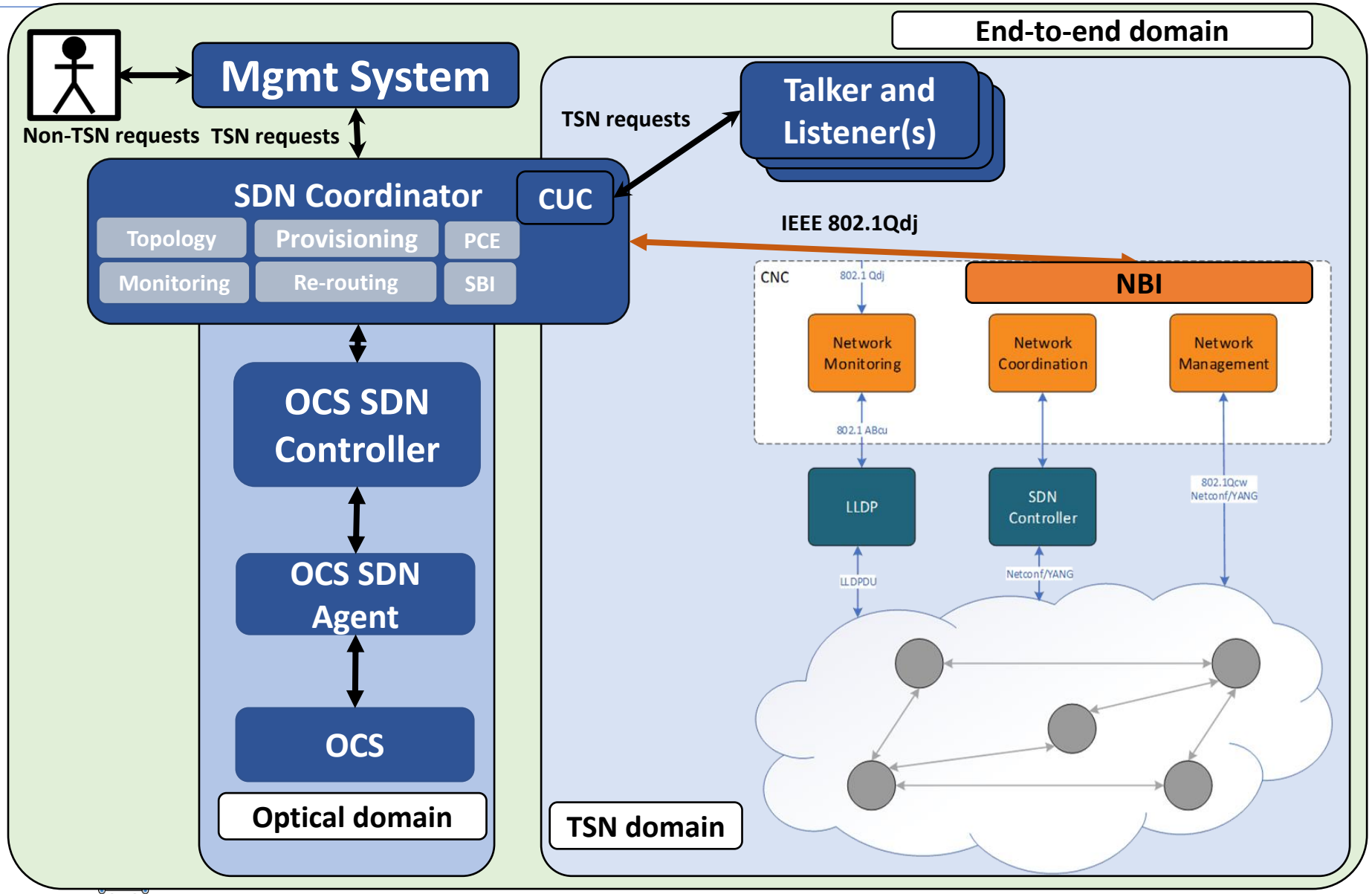
Topology, for managing the NGCO information resource and connectivity

Provisioning, for managing the TSN flows in a multi-domain and multi-technology Transport Network

PCE, for computing the path giving the possible timeliness constraints

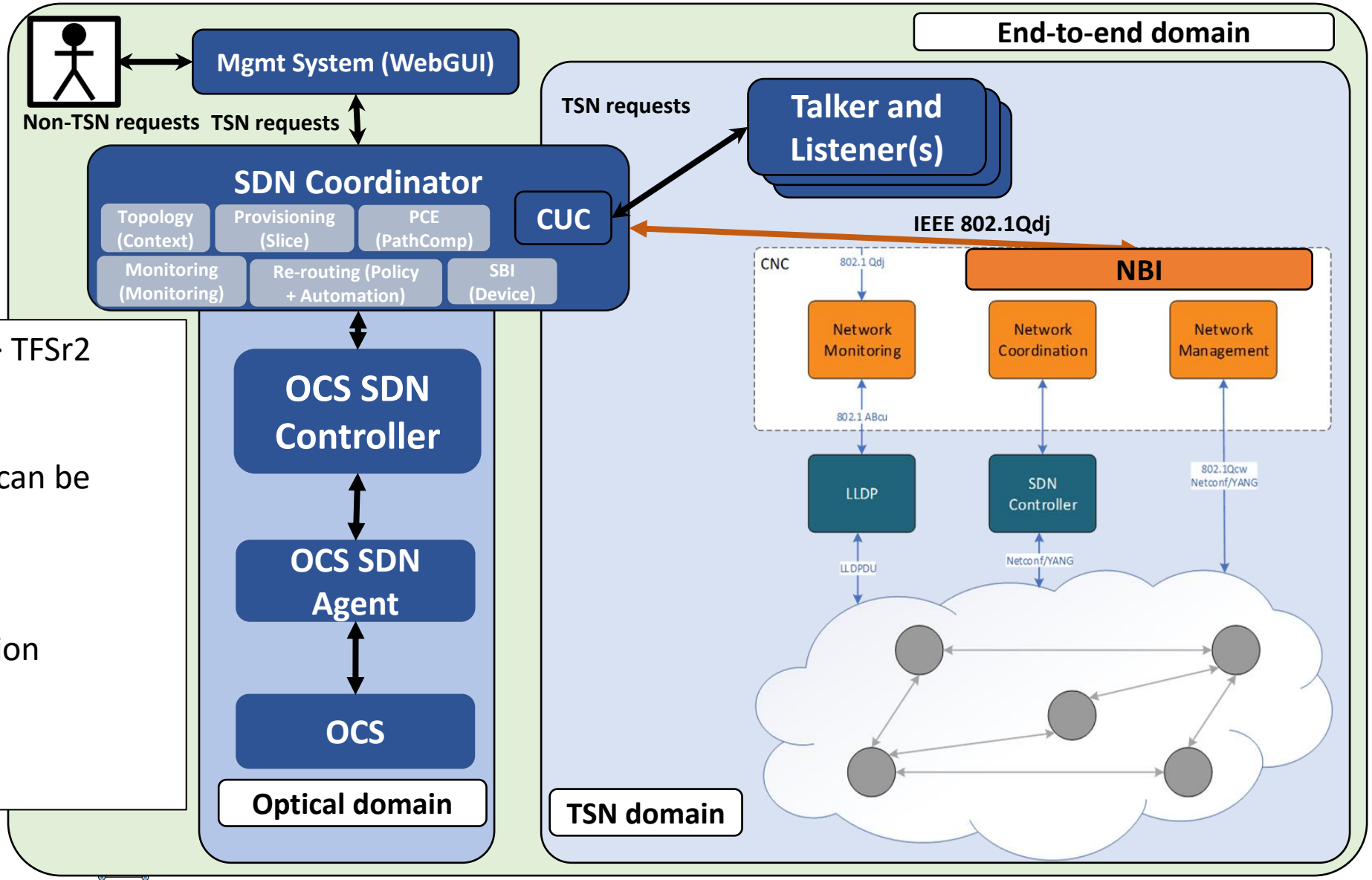
Monitoring + **Rerouting**, for periodically checking the resource status and possibly (and automatically) perform re-routing or load-balancing actions at Up link level

SBI for interacting with the not-TSN (OCS) and TSN domains



SDN Coordinator – Building blocks mapping to TFS components

Internal building blocks of SDN Coordinator can be mapped into one or more SW TFSr2 components:



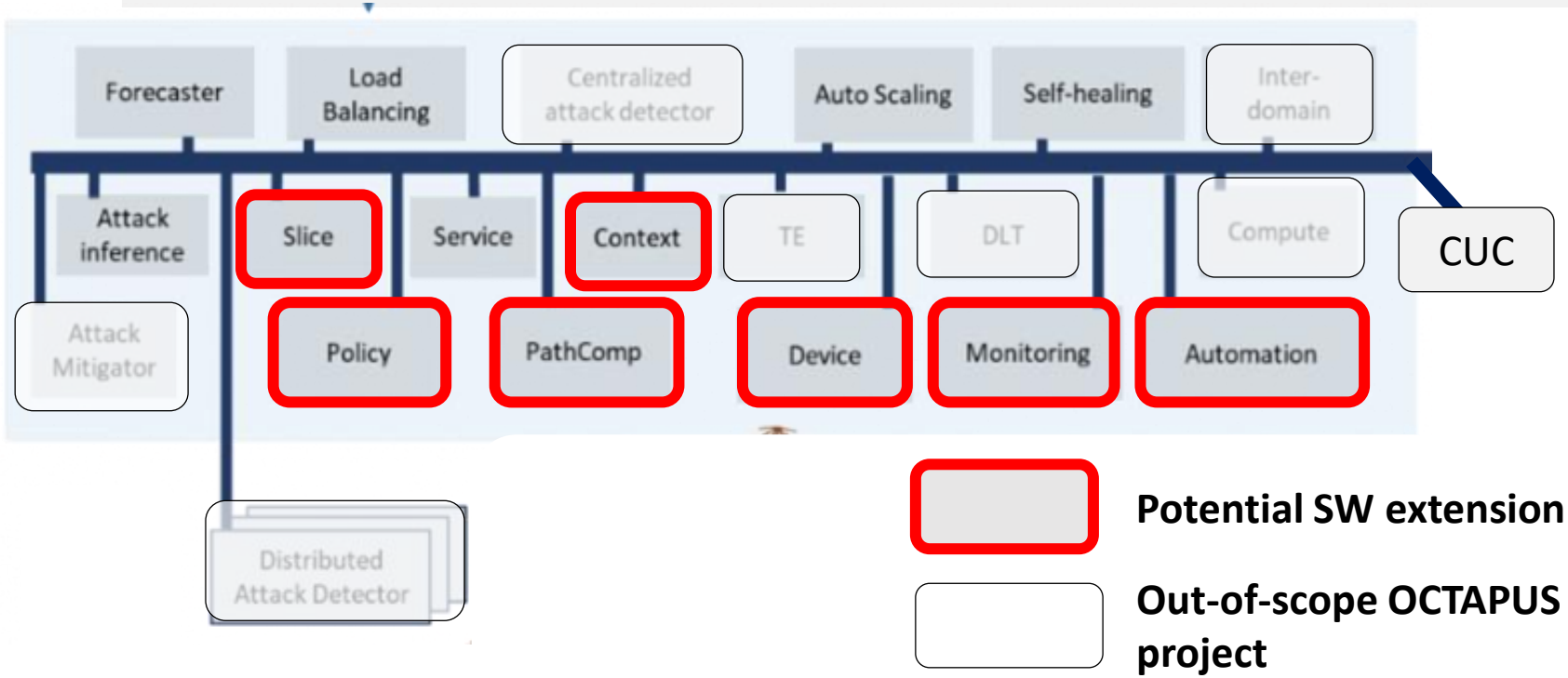
SDN Coordinator building block -> TFSr2 component:

- **Topology** -> Context
- **Provisioning** -> Slice (TSN flow can be considered a slice)
- **PCE** -> PathComp
- **Monitoring** -> Monitoring
- **Re-routing** -> Policy + Automation
- **SBI** -> Device
- **CUC** -> New TFS component
- **Mgmt system ->** WebGUI

TFS in OCTAPUS: components to extend for the SDN Coordinator

Mapping to TFSr2 and Software extensions:

- **Context:** storage of OCS and TSN domains data information (**Topology**) -> Support of OCS and TSN data models
- **Device:** develop plugins for the OCS and the TSN domains **(SBI) ->** OCS NETCONF driver + CNC driver
- **Slice**: Support of non-TSN and TSN flows services LCM in a multi-technologic domain network (**Provisioning**)
- **PathComp**: extension of current path computation algorithm considering the TSN flow constraints (**PCE**)
- **Monitoring**: Periodic collection of information from NGCO (where available) (**Monitoring)**
- **Automation + Policy**: automatic traffic balancing between UPLINK switch made on some rules (**Rerouting**)
- **WebUI (not in the picture)**: some customizations because of specific requests for NGCO Control plane **(Mgmt System)**



Notes:

CUC as new component within TFSr2 ecosystem

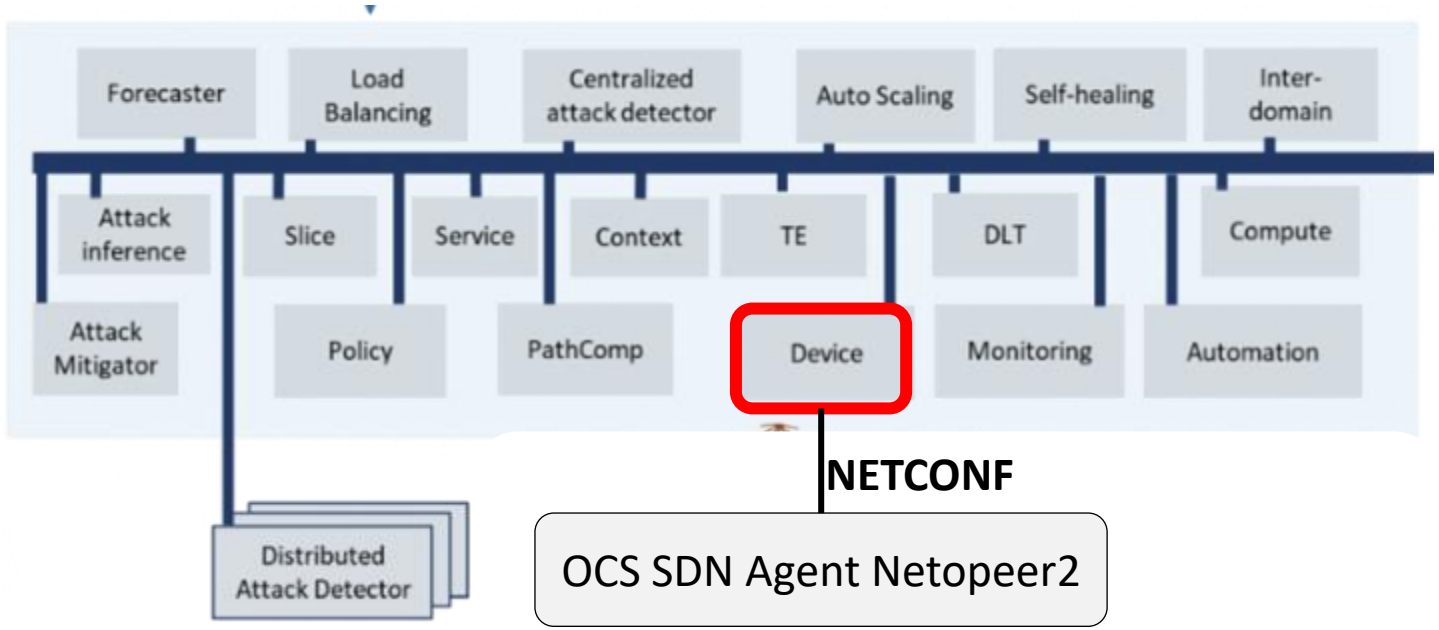
Some TFS components are out-of-the scope of OCTAPUS project

Forecast component might be used in future releases for automated prediction of traffic flows

Initial tests with Netopeer2 SDN Agent

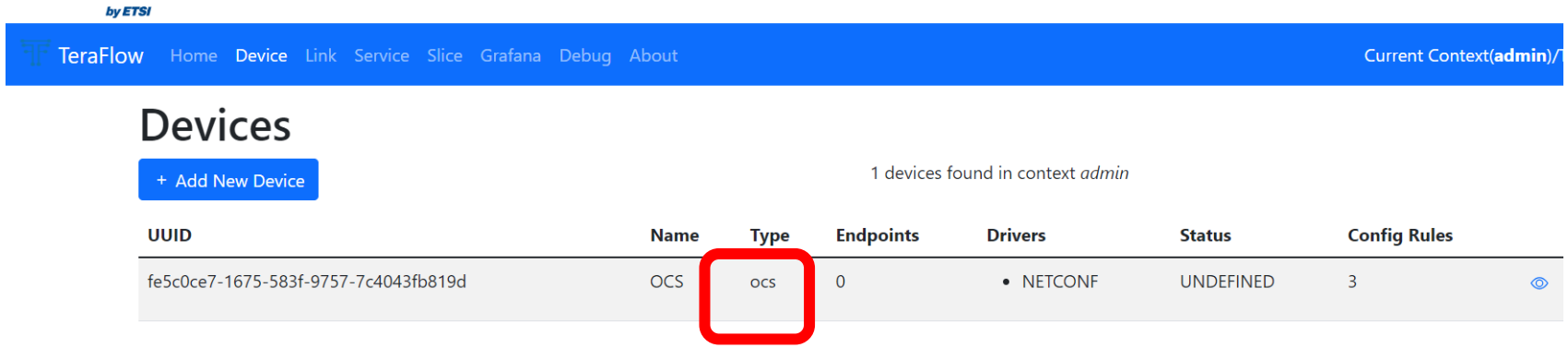
Netopeer2 for emulation of simple network scenarios

- Open-source SDN Agent in C and C++
- Support of basic NETCONF RPC (get-config, edit-config, commit, …)
- Support of candidate and running data stores
- Specific modules generated by YANG models
  - Custom YANG models added for control of OCS nodes



Initial developments and tests with Netopeer2:

- New device type added for OCS management, with OCS data read and configured from/to running datastore
- OCS device available in TFS



Sum up and next steps

Sum up:

- What is OCTAPUS
- NGCO Control plane
- TFSr2 Software modularity exploited to make extensions
  - Support of TSN flow management in a TSN/hybrid network relying on IEEE802.Qdj specification (public draft YANG model as baseline for now)
  - Initial extensions and tests on SBI\Device TFS component started

Next steps:

- **Developments and integration within TFS**
  - Data model extension for supporting TSN based data models -> Create/update *protofiles* and generate Python classes
  - Plugin developments -> Extend SBI/Device component
  - CUC design, development-> Same development pattern of TFS existing component will be adopted

- **Integrations with optical/TSN network and demonstrations**
  - Validation with the optical network
  - Validation with the TSN network
  - Eventually, end-to-end validation (lab trials and final demo)

Thank you for your attention!

Any questions ?