Building a Network Digital Twin for Hidden Failure Detection

Pedro Martinez-Julia, Ved P. Kafle, Hitoshi Asaeda
Network Architecture Laboratory, Network Research Institute
National Institute of Information and Communications Technology (NICT)
{pedro,kafle,asaeda}@nict.go.jp

IEICE
Kenkyukai NV

21 November 2023

Motivation

- Network resilience requires management systems to find actual and potential, exposed and hidden faults (APEHFs) in network services (NS).
  - Actual faults are those found in the current state of the NS.
  - Potential faults are those that can occur if the NS continues operating without change or in the advent of some probable event.
  - Exposed faults are those that can be easily identified, such as a parameter that is—or will be—out of some boundary.
  - Hidden faults are those that cannot be easily identified, such as the congestion in a link of an underlying network that is heavily used by the virtual networks of the NS, or the existence of an unpatched security problem in some element related to the NS.
- These faults, particularly the hidden faults, can cause network outages, which, in turn, causes enormous damage to society.
- We studied the nature of APEHFs to define a mechanism to find them.

Problem Statement

- Analyze the management information related to a NS that is available to find its APEHFs.
- Management information is:
  - NS definitions.
  - NS configuration parameters.
  - NS management metrics—telemetry.
- Management information is produced by the manager of the elements forming the NS—such as a VIM—and other components of a management system.

Proposed Solution (I)

- We propose to analyze the provided management information by an algorithm that first constructs a network digital twin (NDT) of the NS, and then analyzes it using what-if simulation and semantic reasoning to find the APEHFs.
- To build an NDT with high fidelity, our algorithm collects as much information as possible, in the form of knowledge objects (KOs).
- Each KO is built by the producer of the data it contains. We have:
  - NS definition KOs provided by OSM services.
  - NS configuration KOs provided by the VIM—e.g., OpenStack.
  - Telemetry KOs provided by monitoring elements.
- Our algorithm receives all KOs, transmitted through iCPN, using TKDP and TLC, and creates or updates the NDT.

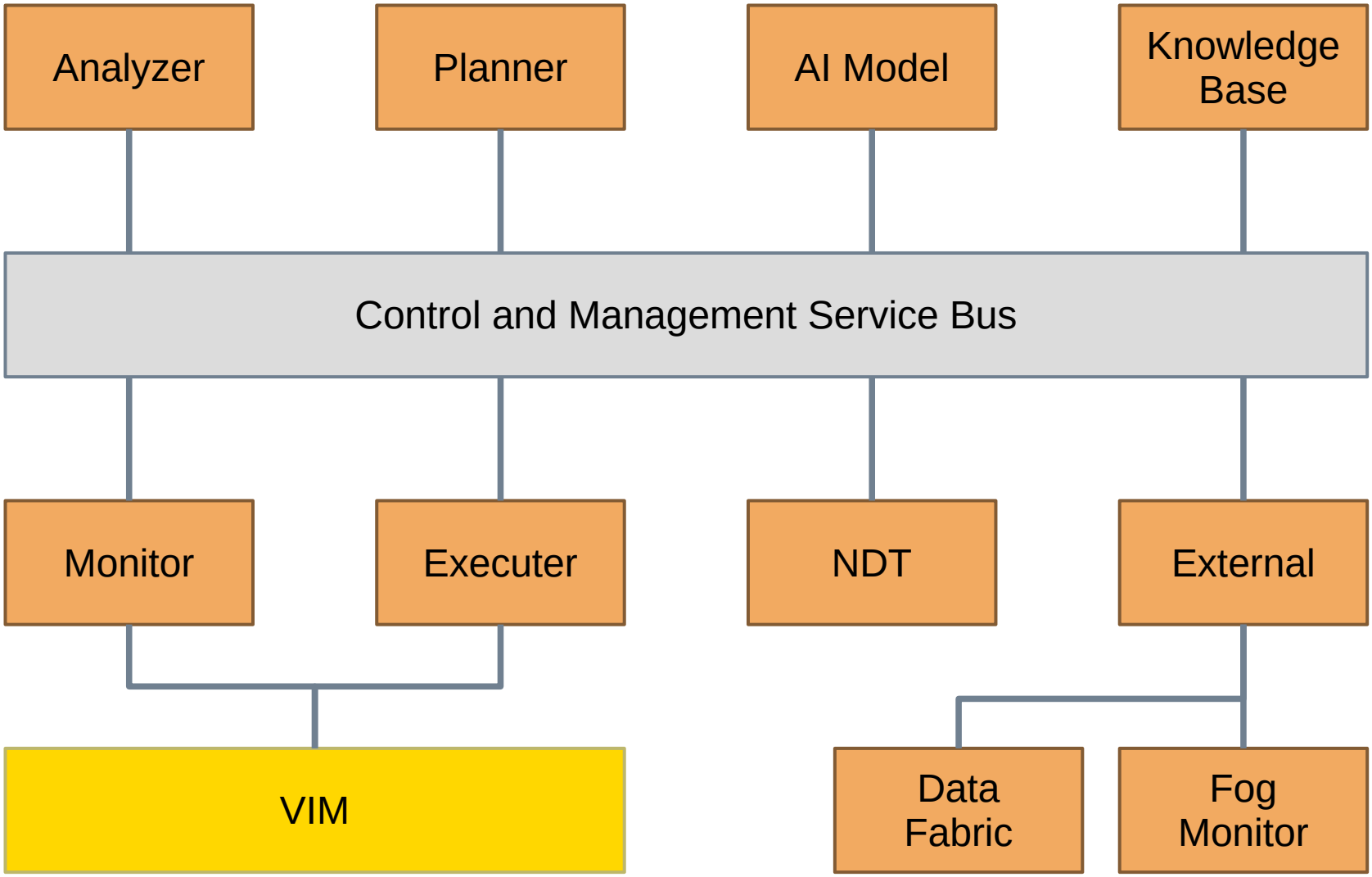
Proposed Solution (II)

- Our algorithm analyzes the resulting NDT—it applies:
  - Several prediction mechanisms to project metrics simultaneously to obtain different what-if situations **represented in the NDT**.
  - Semantic reasoning to **label exposed faults**, either actual and predicted.
  - Analyzing long chains of semantic links that are not related to faults, to find and **label hidden faults**.
- Altogether, all faults form the resulting set of APEHFs.

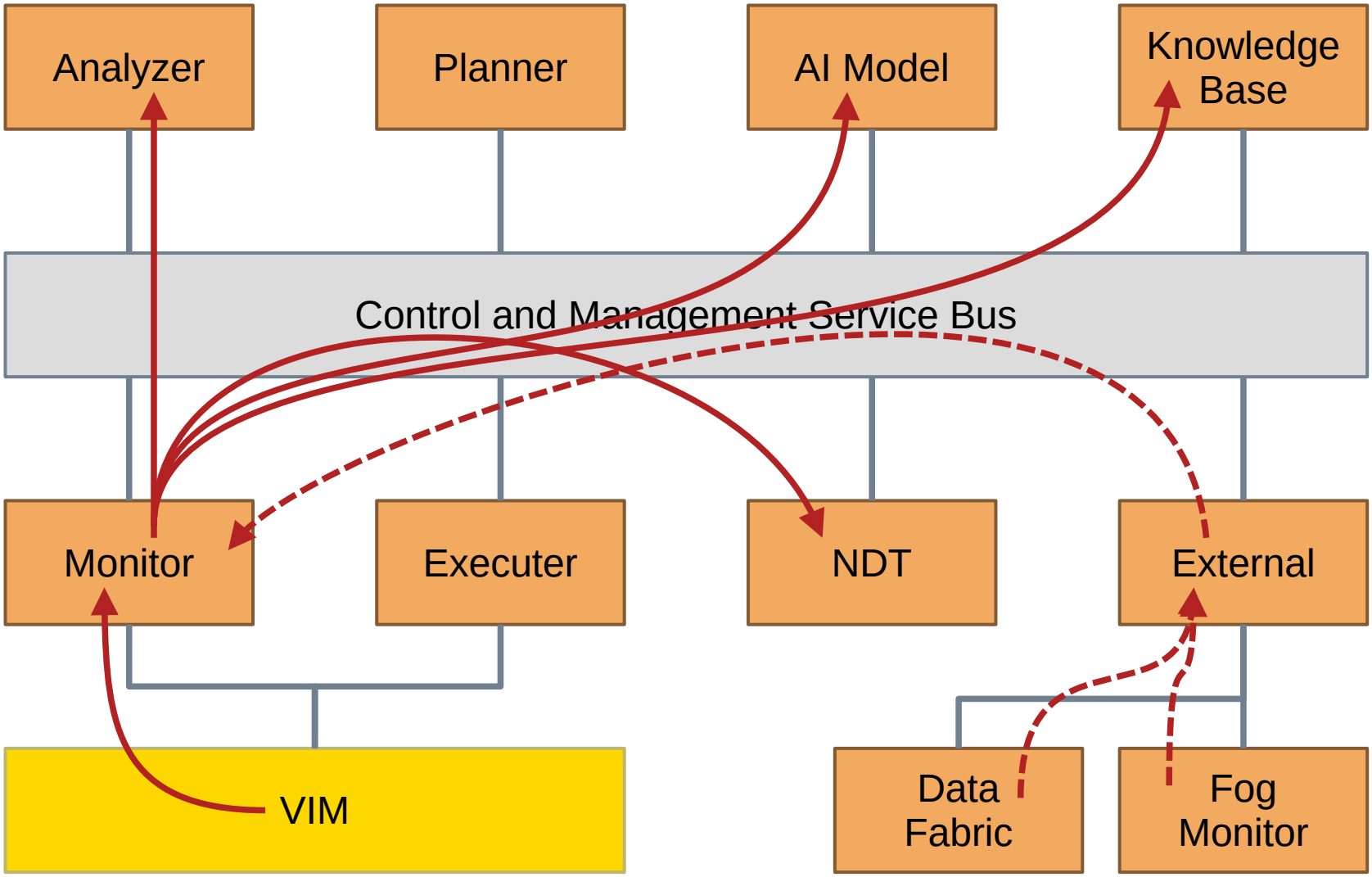
Implementation

- We propose to incorporate the NDT functions to two aligned abstractions:
  - A framework for applying artificial intelligence to network management, named **AINEMA**.
  - The network service automation system (**NSAS**).
- Both **AINEMA** and **NSAS** are implemented in Open-Source Distributed MANO (**OSDM**), which is an evolution of Open-Source MANO (**OSM**).

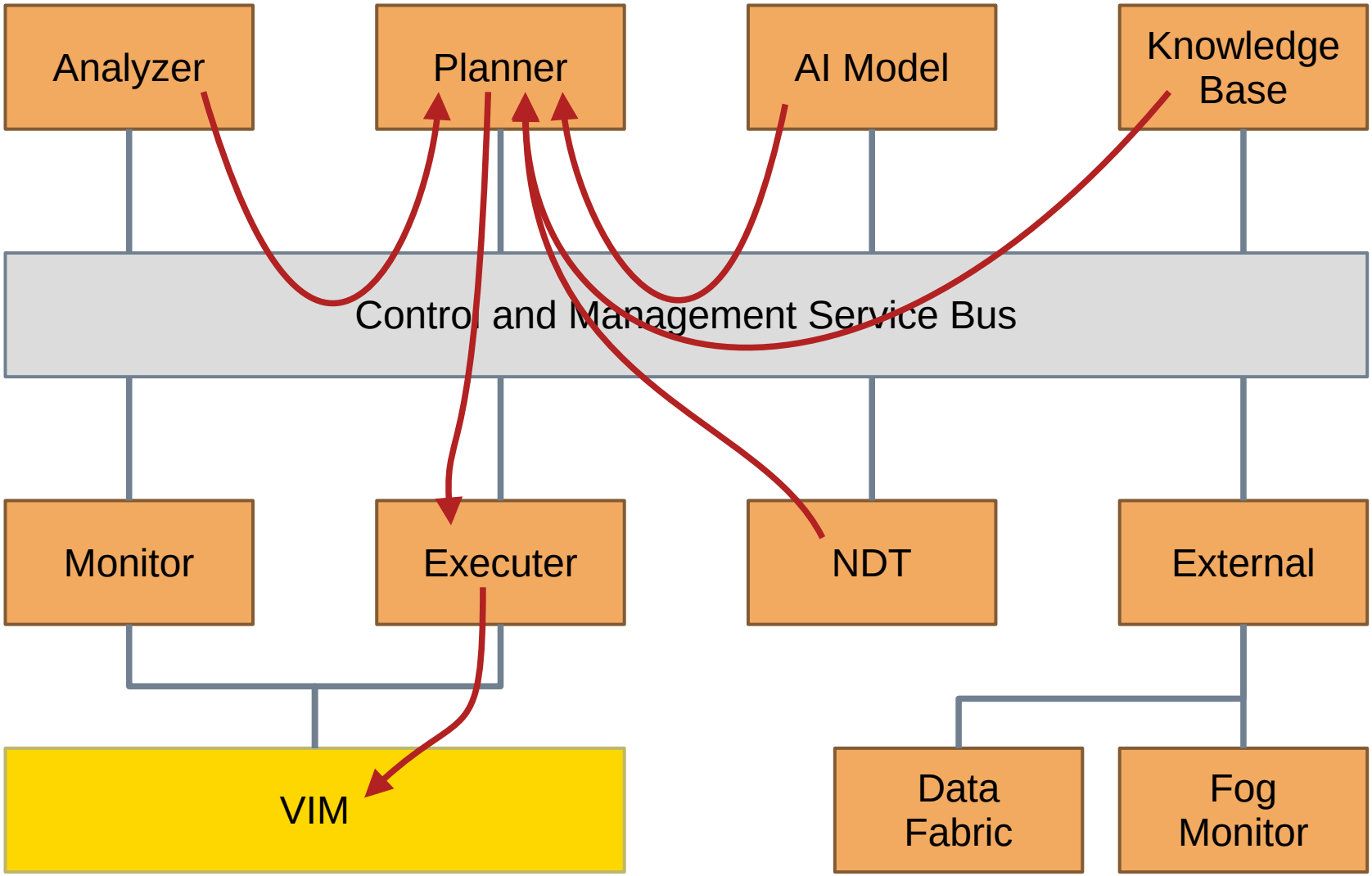
AINEMA → NSAS: Overview



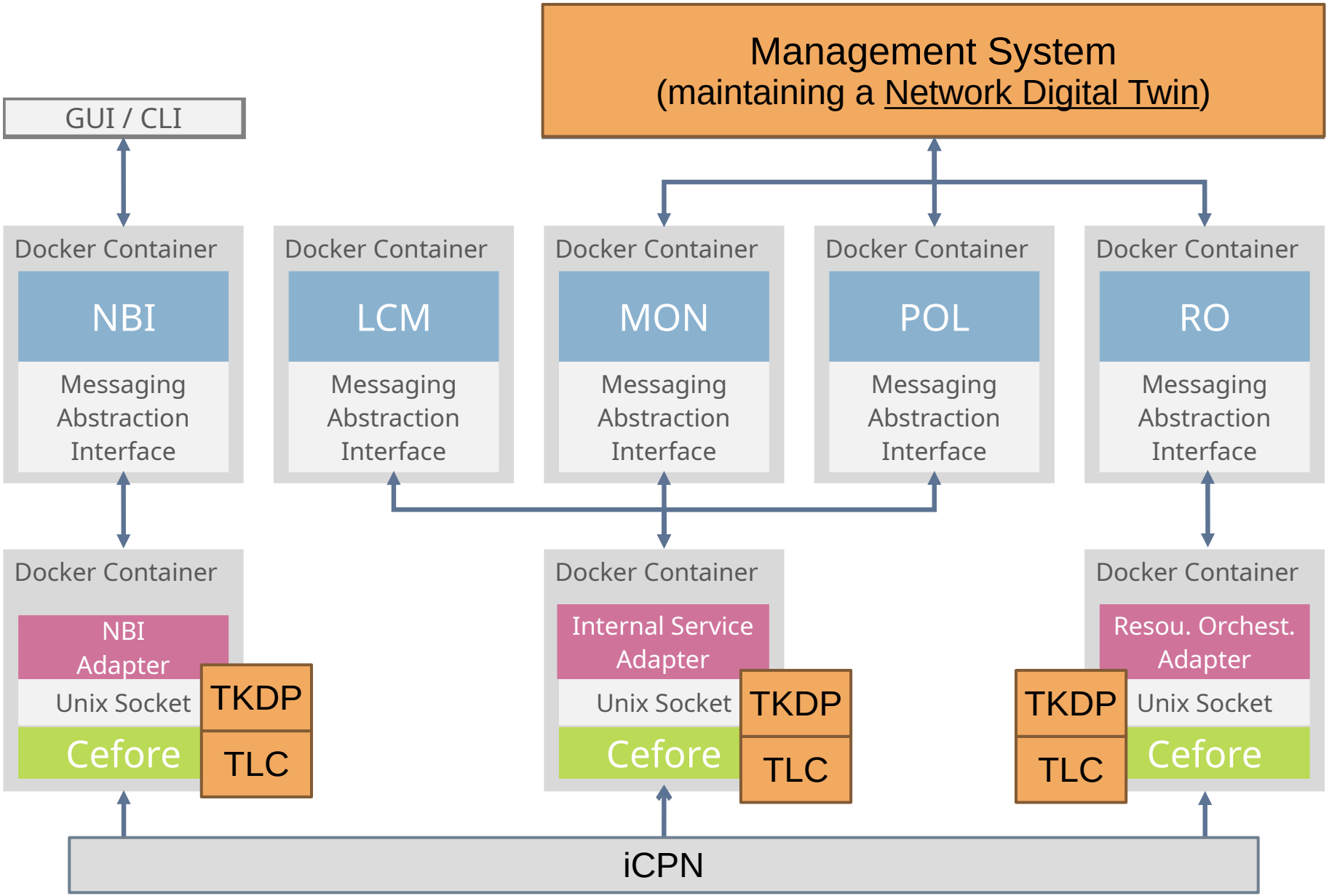
AINEMA → NSAS: Operation (I)



AINEMA → NSAS: Operation (III)



AINEMA → NSAS → OSDM: Deployment (I)



AINEMA → NSAS → OSDM: Deployment (II)

- We implemented the architecture as the Open Source Distributed MANO (OSDM), which has:
  - An adapter that uses Cefore underneath to provide iCPN—the ICN-based control plane network.
  - The TLC component that provides support for lossy/lossless compression to the adapters of other components.
  - The TKDP components that provides support for retrieving processed information from monitored elements in the form of knowledge objects.
- All components are connected to message abstraction interfaces.
- Cefore engines cache data in the intermediate network elements and deliver them efficiently to all requesters, such as the OSM modules.

Conclusions

- We were motivated by the need of finding APEHFs from available information.
- We defined a method to construct an NDT, a method to find APEHFs, and two abractions required to implement them in network management systems.
- We implemented in OSDM of our view of the NDT and our method to find APEHFs.
- We are evaluating our proposal through experiments in a real platform.

Thanks for Your Attention

Questions?

– EOF –