Software Defined Infrastructure

The FELIX architecture blueprint and implementation experience

Kostas Pentikousis

eict

IEEE SDN Initiative Teleconference

22 June 2015

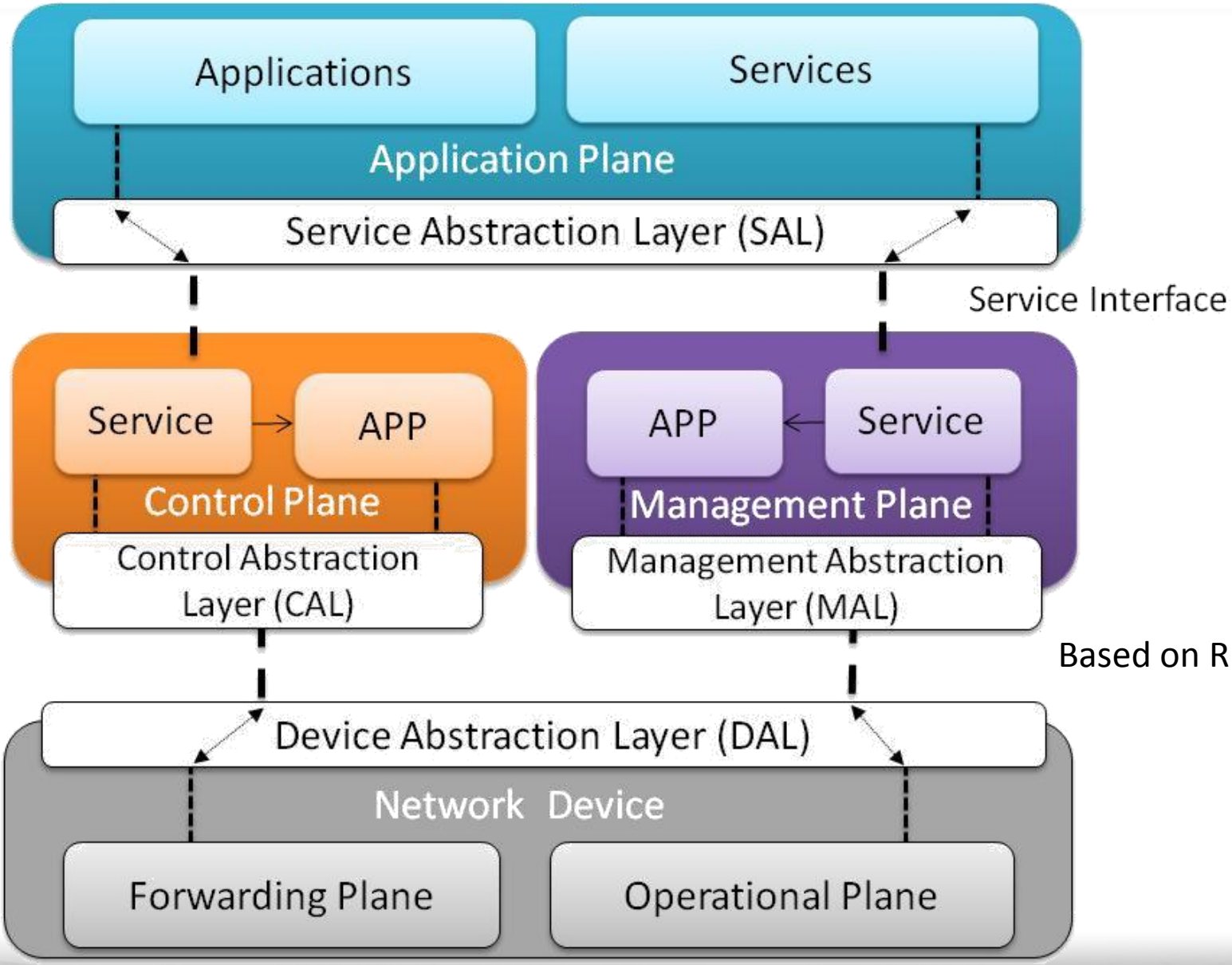
Talk Outline

- Introduction
  - Programmable Networks Paradigm
  - SDN Layers and Architecture Terminology
    - Comparison with ITU-T Y3300 and ONF Architecture
- Toward SDNIs
  - RFC 7426 Applications
    - ALIEN HAL and examples
    - The cooperating layered architecture for SDN (CLAS)
    - SDN-enabled NFV
  - FELIX SDN Experimental Facility
    - Project overview
    - Concepts and Use Cases
    - Architecture and Implementation
- Summary

Programmable Networks Paradigm [1]

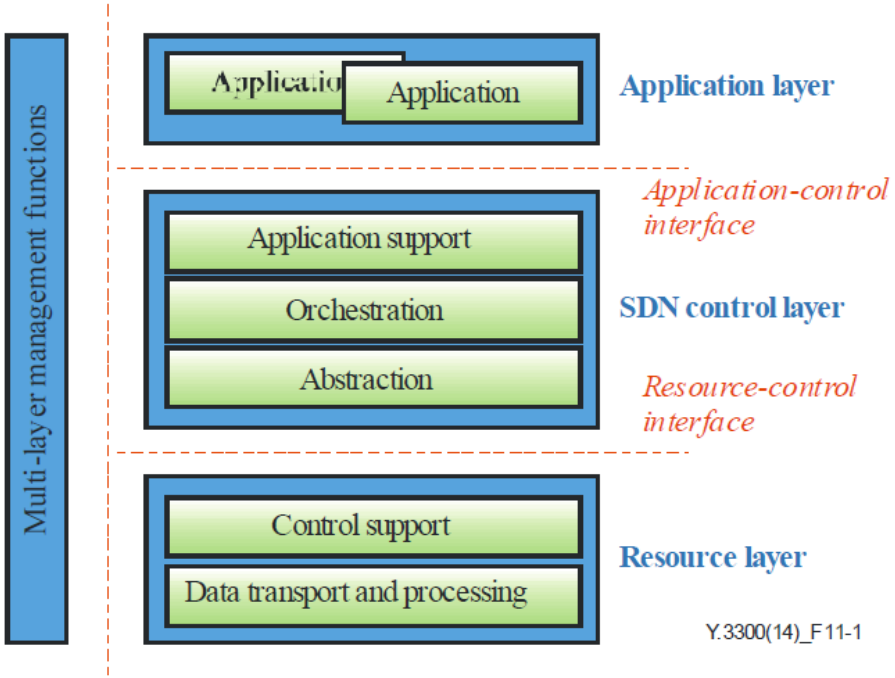
- Active Networks [late 1990s]
  - Networks where devices can perform custom computation on packets
  - Followed models such as
    - *packet capsules* and
    - *programmable network devices*
  - Offered contributions such as service composition and demultiplexing
  - Examples include: *ANTS (Java), SwitchWare (Scripting), NetScrip*, etc.
- Control & Data Plane separation [early 2000s]
  - Speedier innovation due to control- and data-plane decoupling
  - Enable network-wide view which made it easier to infer network behavior
  - Examples include *RCP, ForCES, PCE, Tempest,* but also to some degree EPC
- OpenFlow and Network Operating Systems (NOSs) [late 2000s]
  - Defines a standard interface between the control- and the data-plane that builds on existing hardware without needs for customization
  - NOSs such as *Onix*, *ONOS*, etc. were introduced to provide an abstraction layer between network state awareness and control logic
  - Introduction of distributed-state management for logically-centralized control

SDN Layers and Architecture Terminology [2]

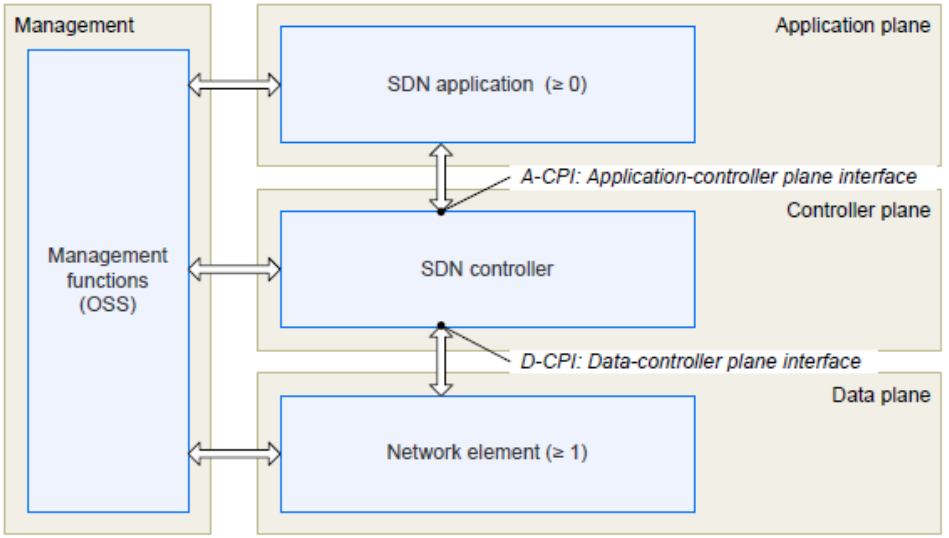


Based on RFC 7426

ITU-T Y.3300 and ONF TNR-502



Source: ITU-T Y.3300



Source: ONF TNR-502

Network management is everything you don't know how to do.

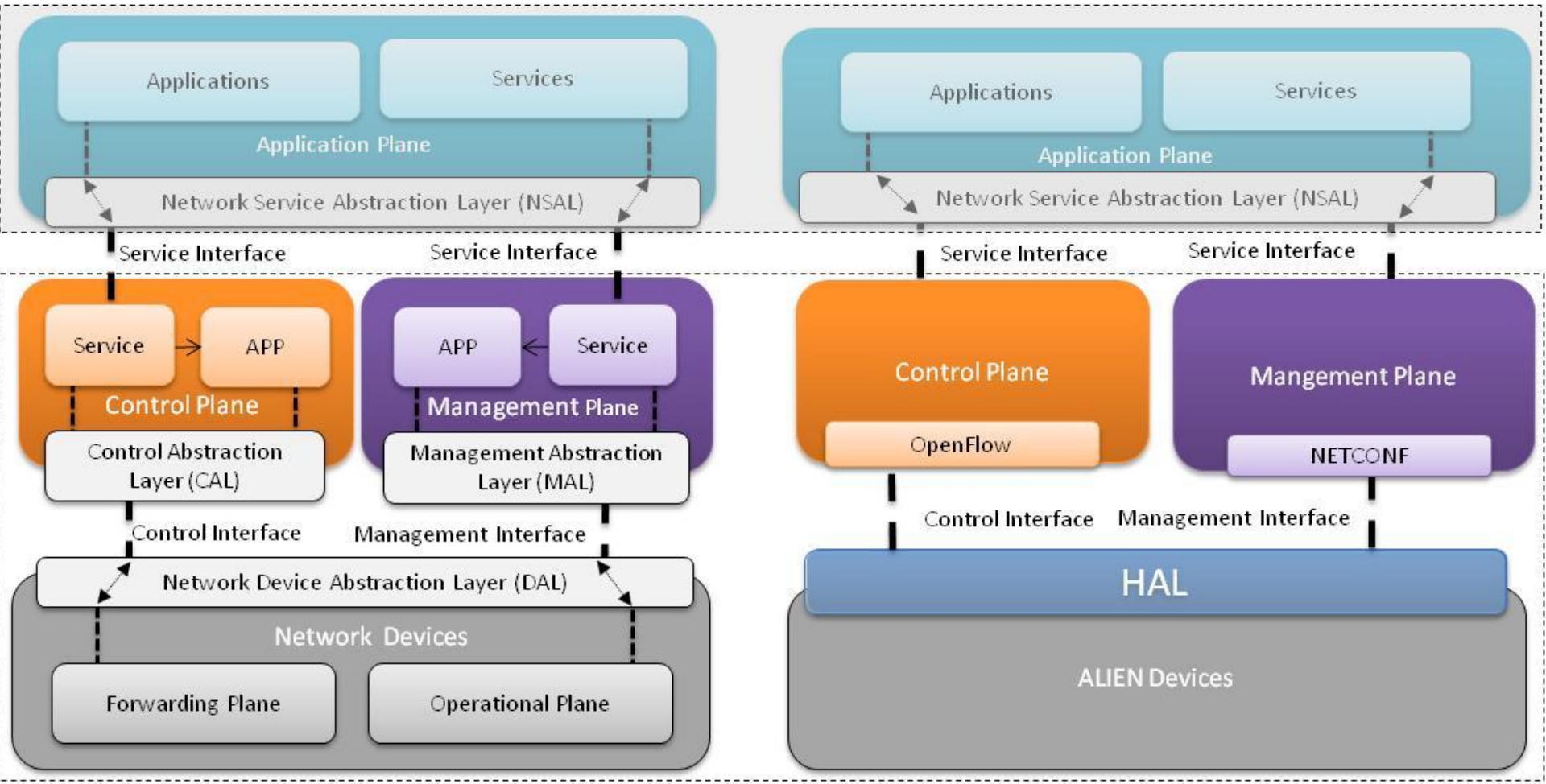
If you knew how to do it, you would have put it in the base system, and then you wouldn't have called it network management.

–Dave Oran, Cisco Fellow

Talk Outline

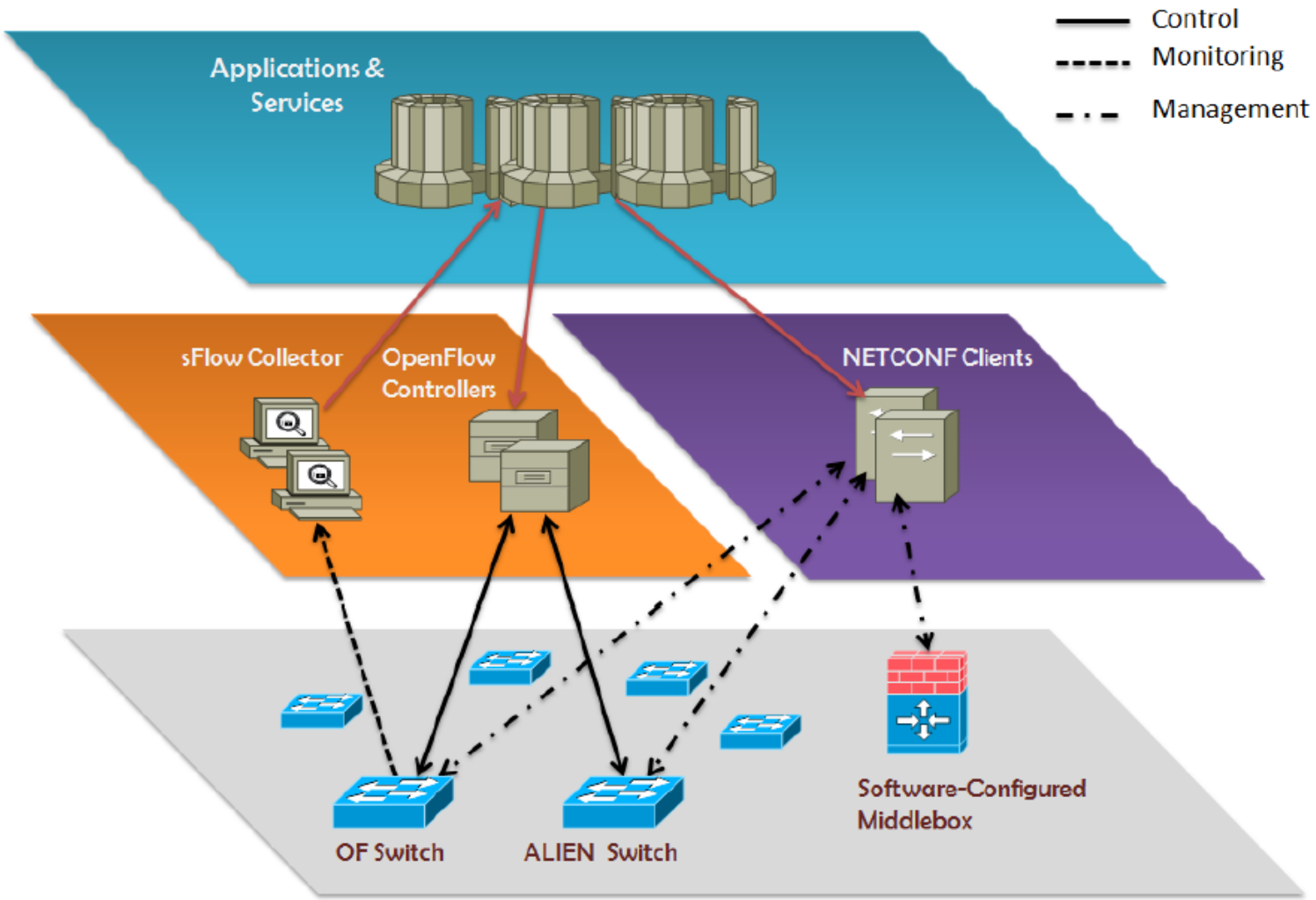
- Introduction
  - Programmable Networks Paradigm
  - SDN Layers and Architecture Terminology
    - Comparison with ITU-T Y3300 and ONF Architecture
- Toward SDNIs
  - RFC 7426 Applications
    - ALIEN HAL and examples
    - The cooperating layered architecture for SDN (CLAS)
    - SDN-enabled NFV
  - FELIX SDN Experimental Facility
    - Project overview
    - Concepts and Use Cases
    - Architecture and Implementation
- Summary

RFC 7426 Applications: ALIEN HAL [3]



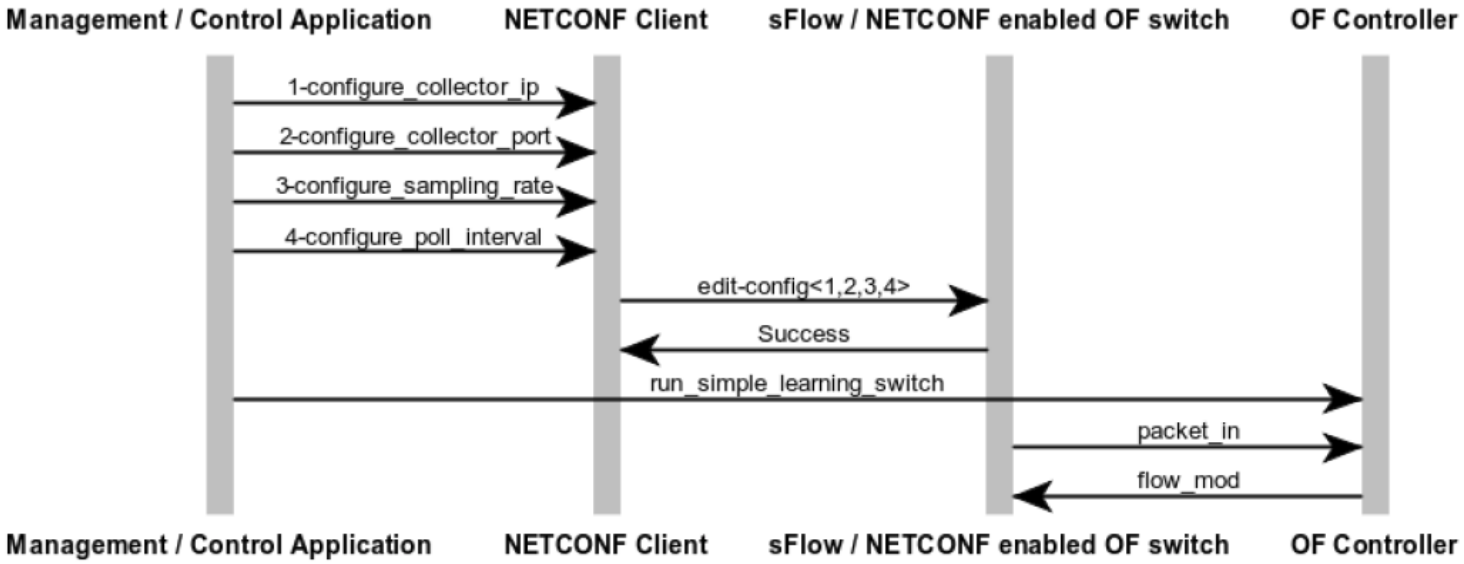
Source: [1]

RFC 7426 Applications: Toy Example (1/2)



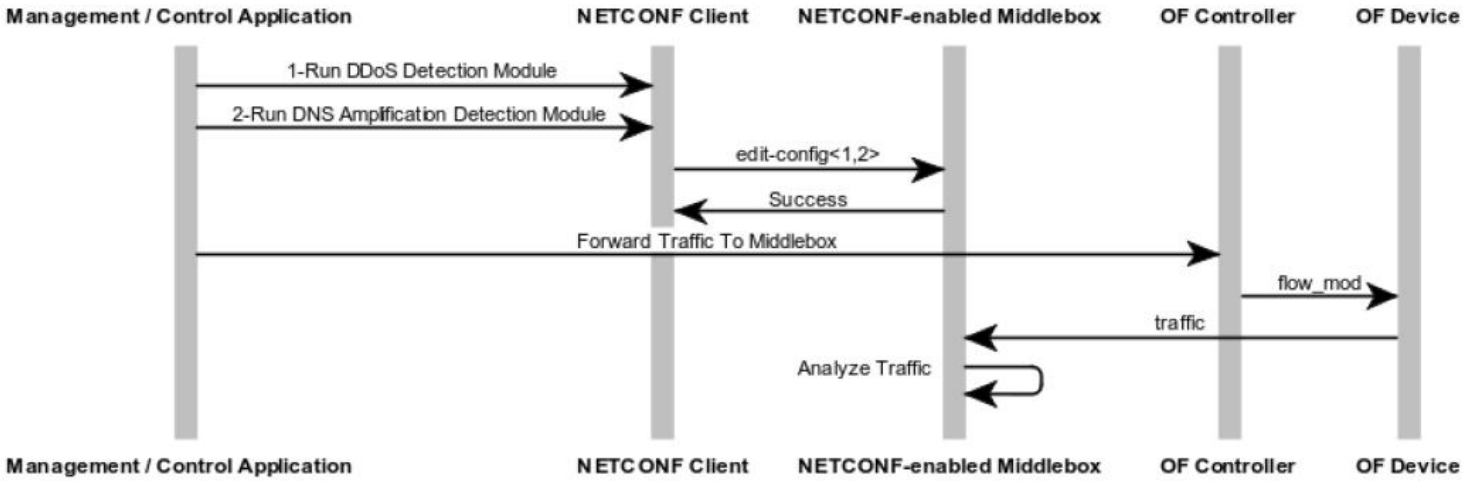
Source: [1]

RFC 7426 Applications: Toy Example (2/2)



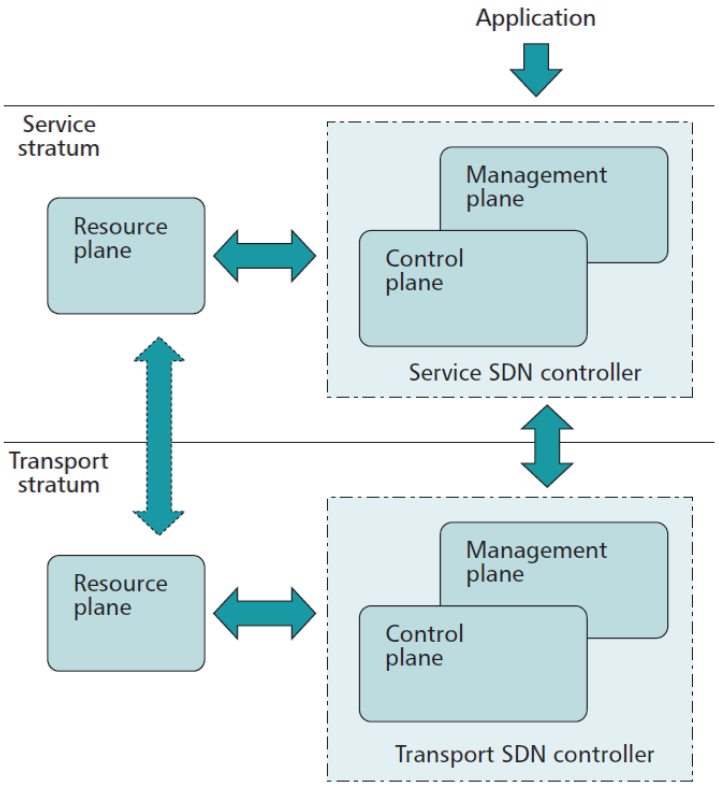
Network Monitoring

Network Security

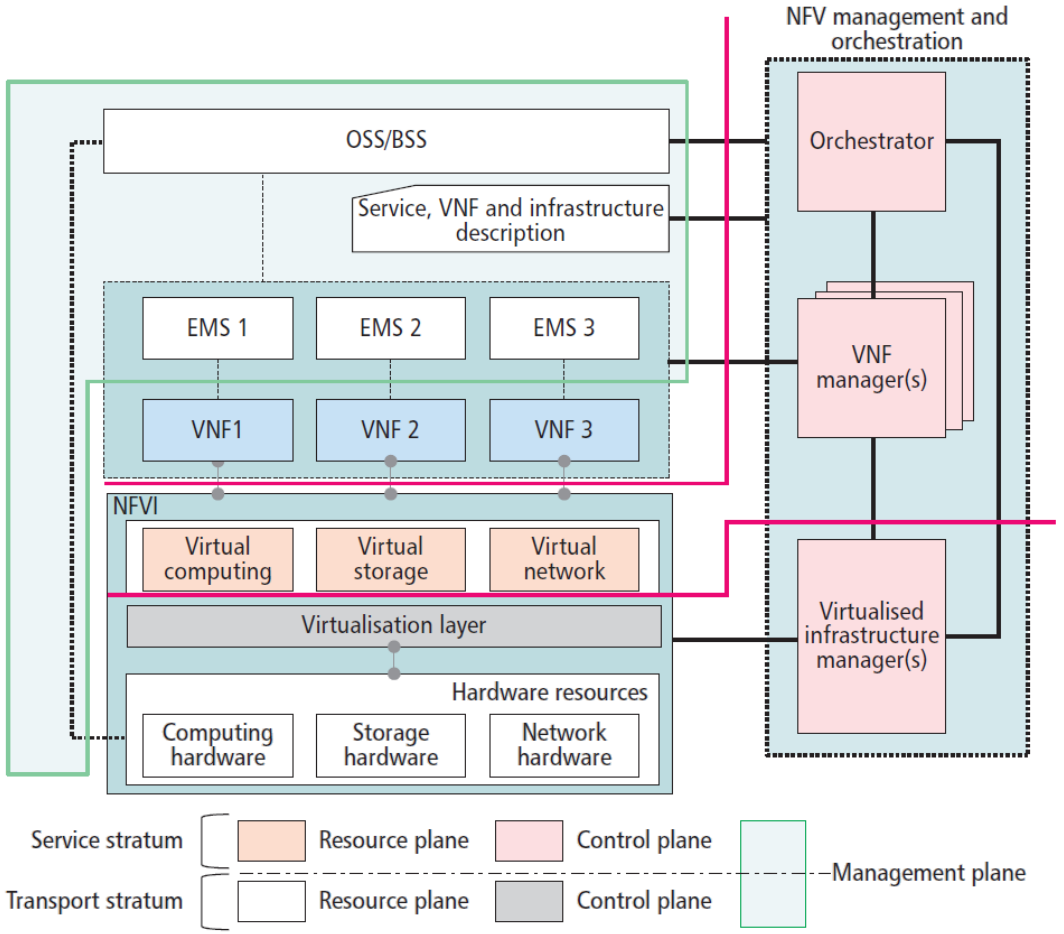


Source: [1]

RFC 7426 Applications: CLAS

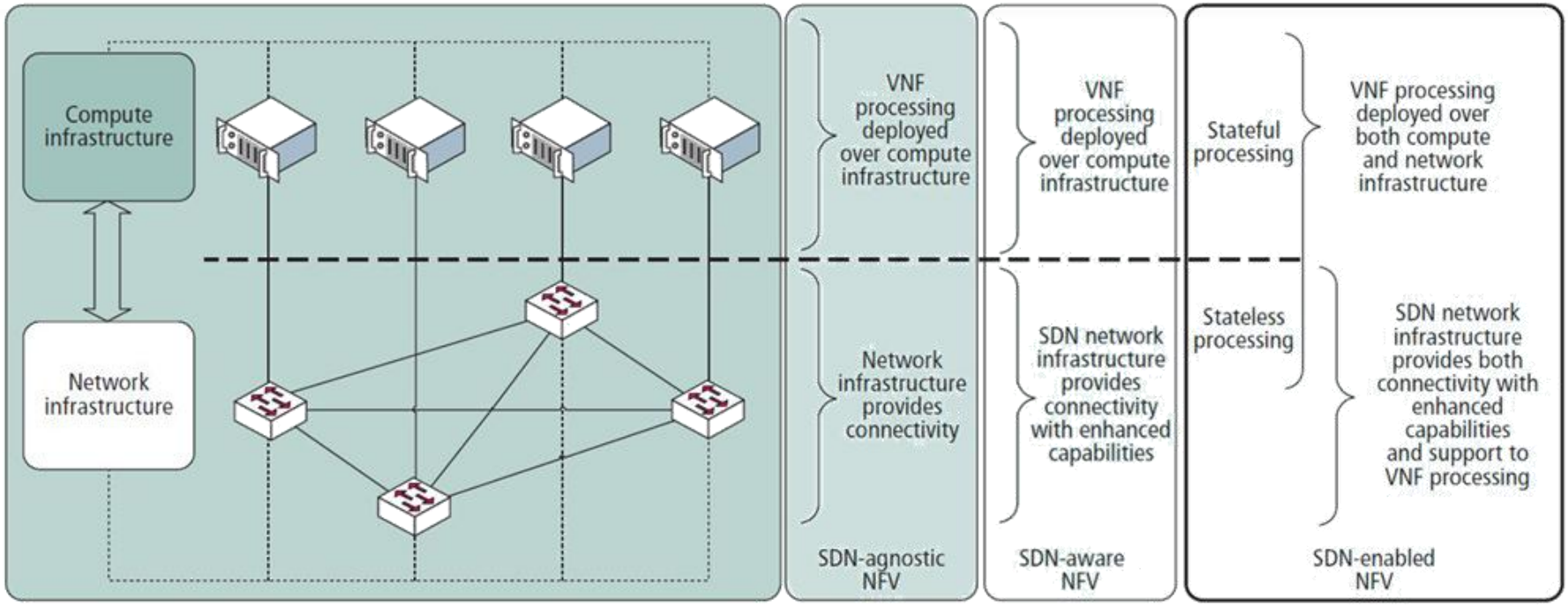


Source: Cooperating Layered Architecture for SDN (CLAS) [4]

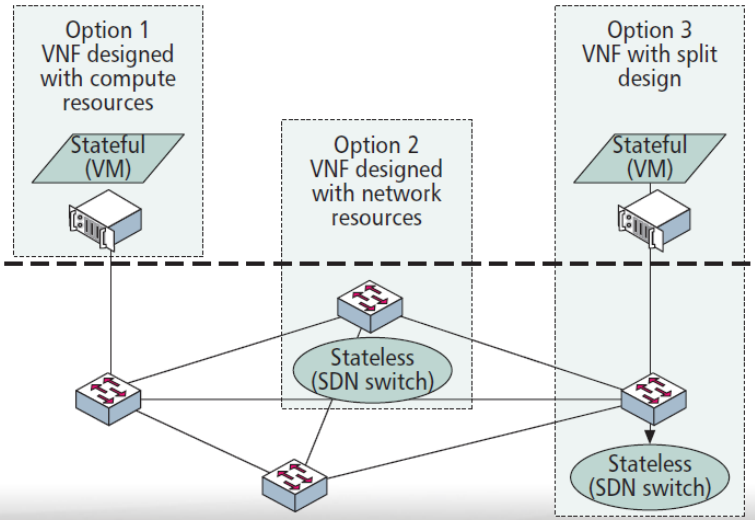


Source: Layered approach to NFV control [4]

RFC 7426 Applications: SDN-enabled NFV



Source: NFV architecture evolution [5]



Talk Outline

- Introduction
  - Programmable Networks Paradigm
  - SDN Layers and Architecture Terminology
    - Comparison with ITU-T Y3300 and ONF Architecture
- Toward SDNIs
  - RFC 7426 Applications
    - ALIEN HAL and examples
    - The cooperating layered architecture for SDN (CLAS)
    - SDN-enabled NFV
  - FELIX SDN Experimental Facility
    - Project overview
    - Concepts and Use Cases
    - Architecture and Implementation
- Summary

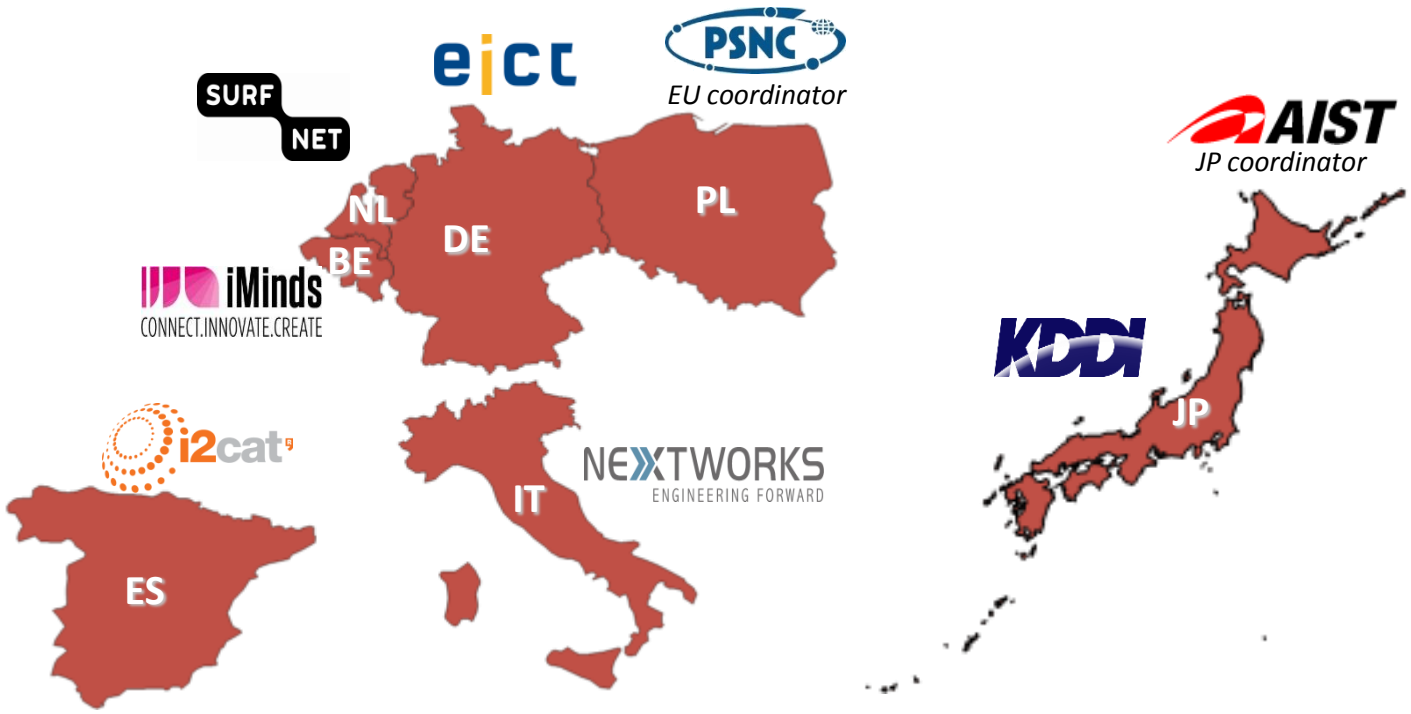
FELIX in a Nutshell

Facts

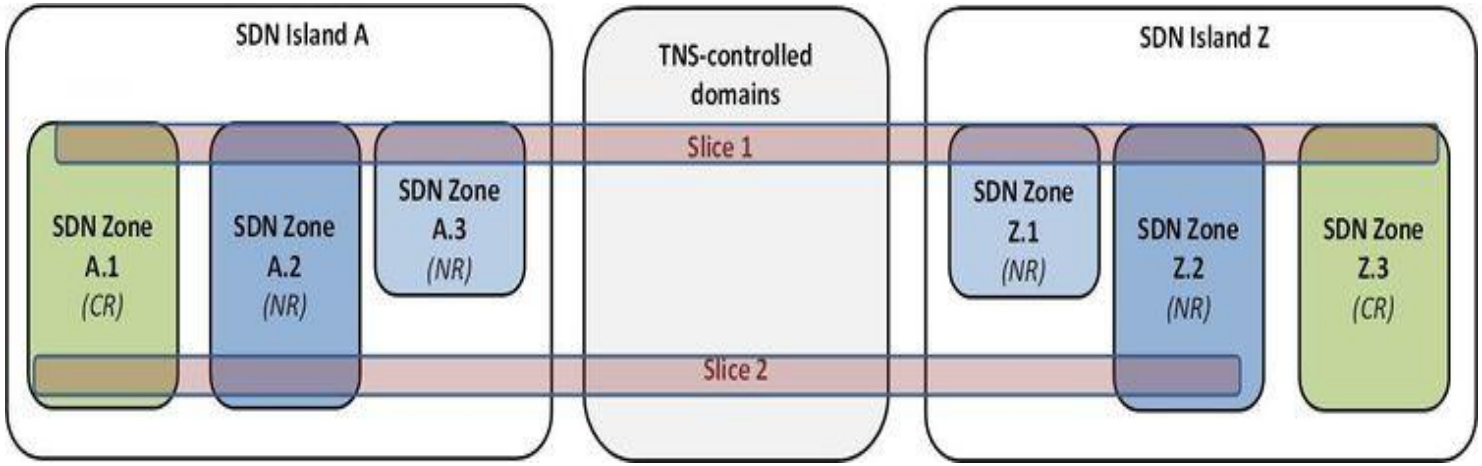
- EC (EU), MIC & NICT (JP) collaborative project
- Project running from April 2013 till March 2016
  - 302 PMs of effort

Objectives

- To create a large-scale testbed federated across two continents
- To define a common software architecture for testbeds

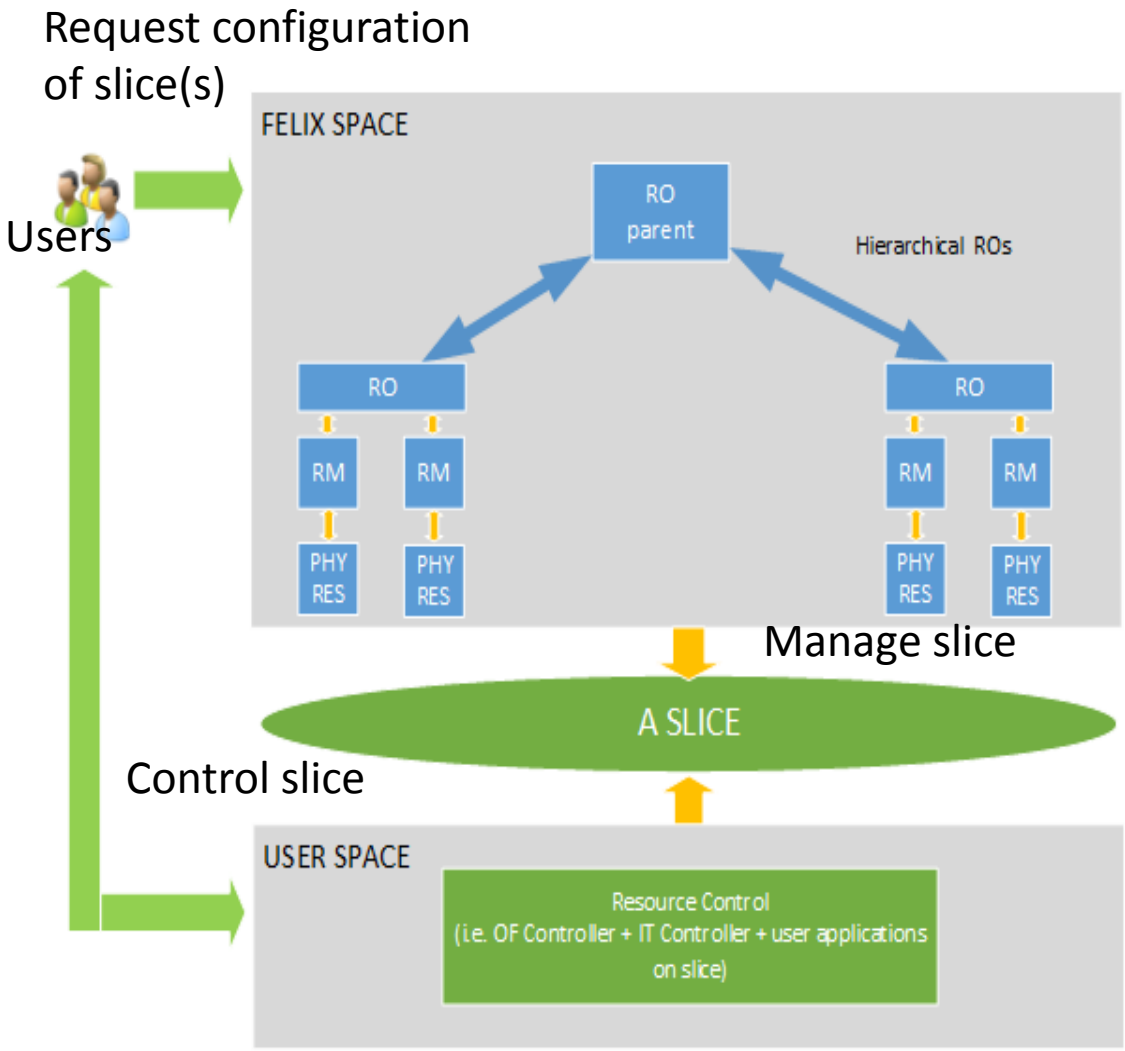


FELIX Concepts



- The **slice** concept is adopted in FELIX
  - Experimental facilities to be provided dynamically on top of the FELIX physical infrastructure (federated testbeds)
- All the experimental facilities are controlled programmatically
  - Facilities are composed of computing and network resources (*CR* and *NR*) belonging to distributed SDN islands in FELIX infrastructure
  - Orchestrate resources in a multi-domain environment
  - In a slice, facilities are interconnected via TN service-controlled domains (transit network)
- User has access and control of a provided slice

FELIX Architecture Overview



- The **FELIX Space** provides users with slices for their own use. Users request slices to an RO.
  - RO: Resource Orchestrator
  - RM: Resource Manager
  - PHY RES: physical resources (testbed)
- The **User Space** consists of any tools and applications that a user wants to deploy to control a slice or execute particular operations

FELIX Use Cases [6]

Two major divisions of use cases

- **Data Domain** use cases

  Optimize the use of the interconnectivity between testbeds to **realize data transfer**

  1. *Data on Demand – delivery of distributed data by setting data flows*
  2. *Pre-processing and delivery of nearly real-time [satellite] data*
  3. *High quality media transmission over long-distance networks*

- **Infrastructure Domain** use cases

  Optimize the use of the infrastructure as a whole, including the **migration of entire data processing workloads**

  4. *Data mobility service by SDN technologies*
  5. *Follow the Sun/Moon – green energy in data centers*
  6. *Disaster recovery by migrating IaaS to a remote data center*

Data Mobility Service by SDN Technologies (Inter-Cloud use case)

A user of a service provided by a cloud system moves to a remote location

How does cloud system monitor performance and "move" selected data closer to the remote location?

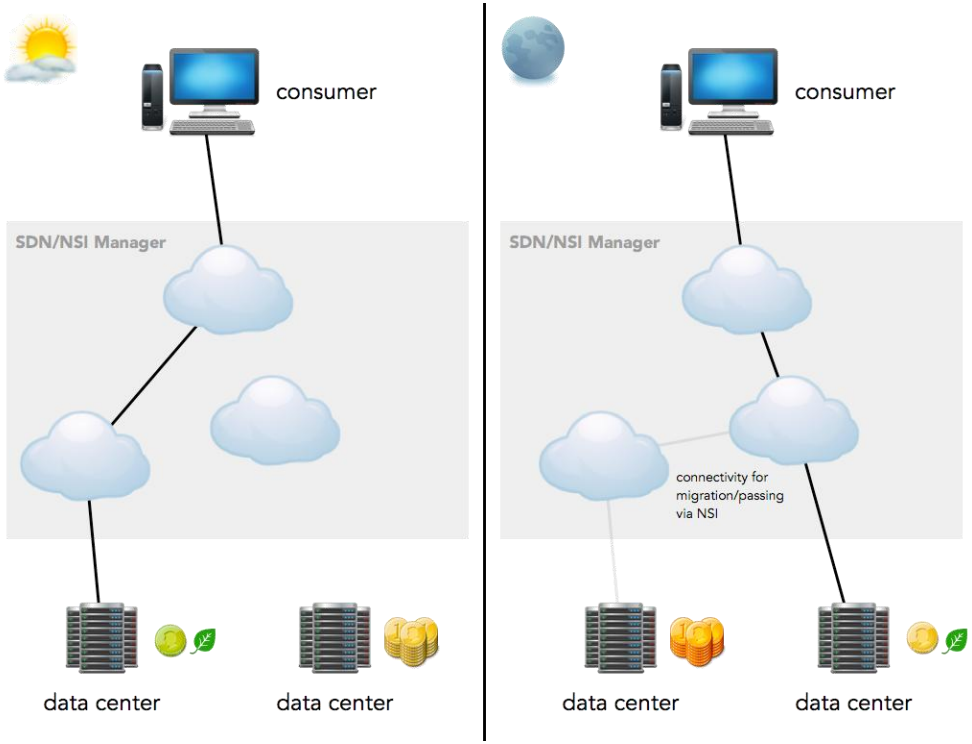
- **Key entity**: User experience monitoring and manager
- **Key solution aspects:** Monitoring, detection of the suitable location, NSI and OpenFlow managers, resource management

Follow the Sun/Moon

Internet usage typically follow a similar daily pattern around the world

How can we move the compute workflow to a suitable data center in the federation?

- **Key entity**: Workload control and management
- **Key solution aspects**: Monitoring, OpenFlow & NSI managers for dynamic, network connections and resource management for (re-)provisioning



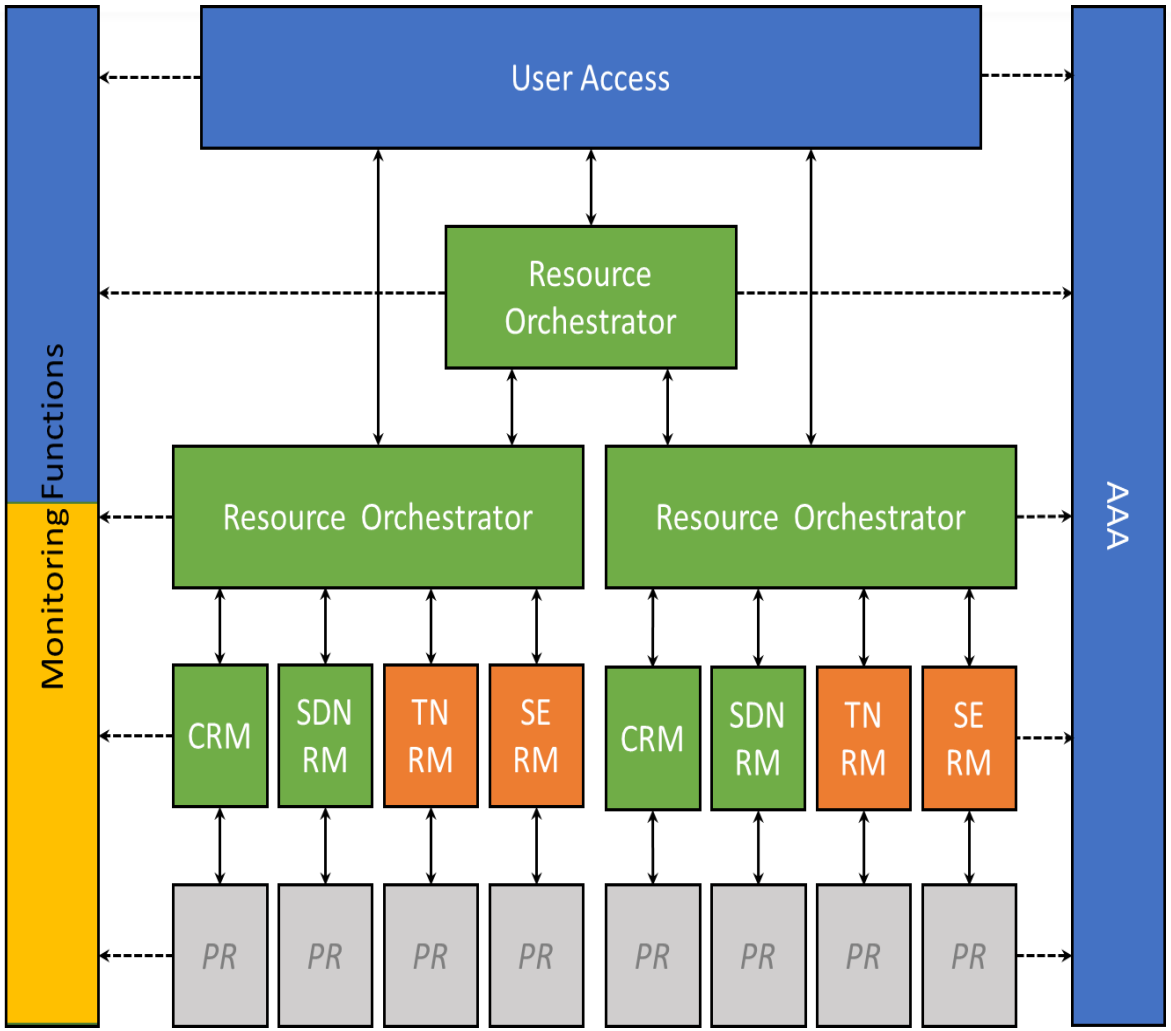
FELIX Architecture [7]

D3.1
Resource Planning and Provisioning

D3.2
Slice Monitoring

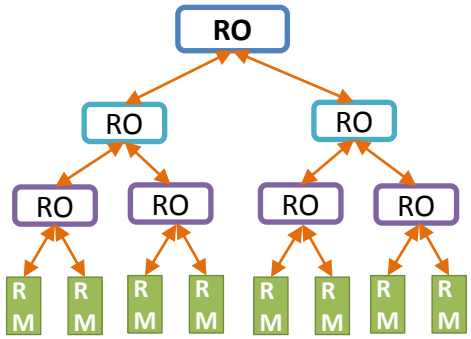
D3.3
Inter-domain networking between SDN slices

D3.4
End User Tools and API

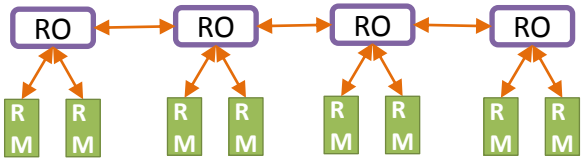


FELIX technical documents/deliverables and architecture whitepaper available at www.ict-felix.eu

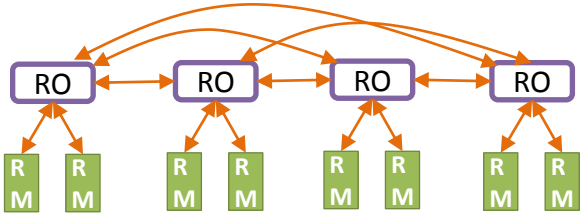
FELIX Architecture: RO Hierarchical Structure Options



Centralized



Distributed

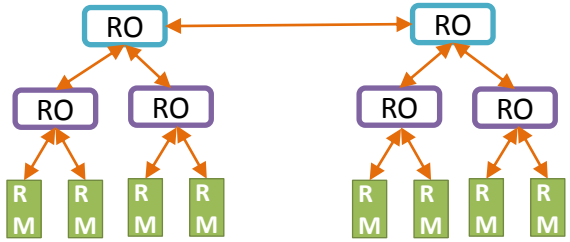


Full mesh

RO Global RO

RO Continent RO

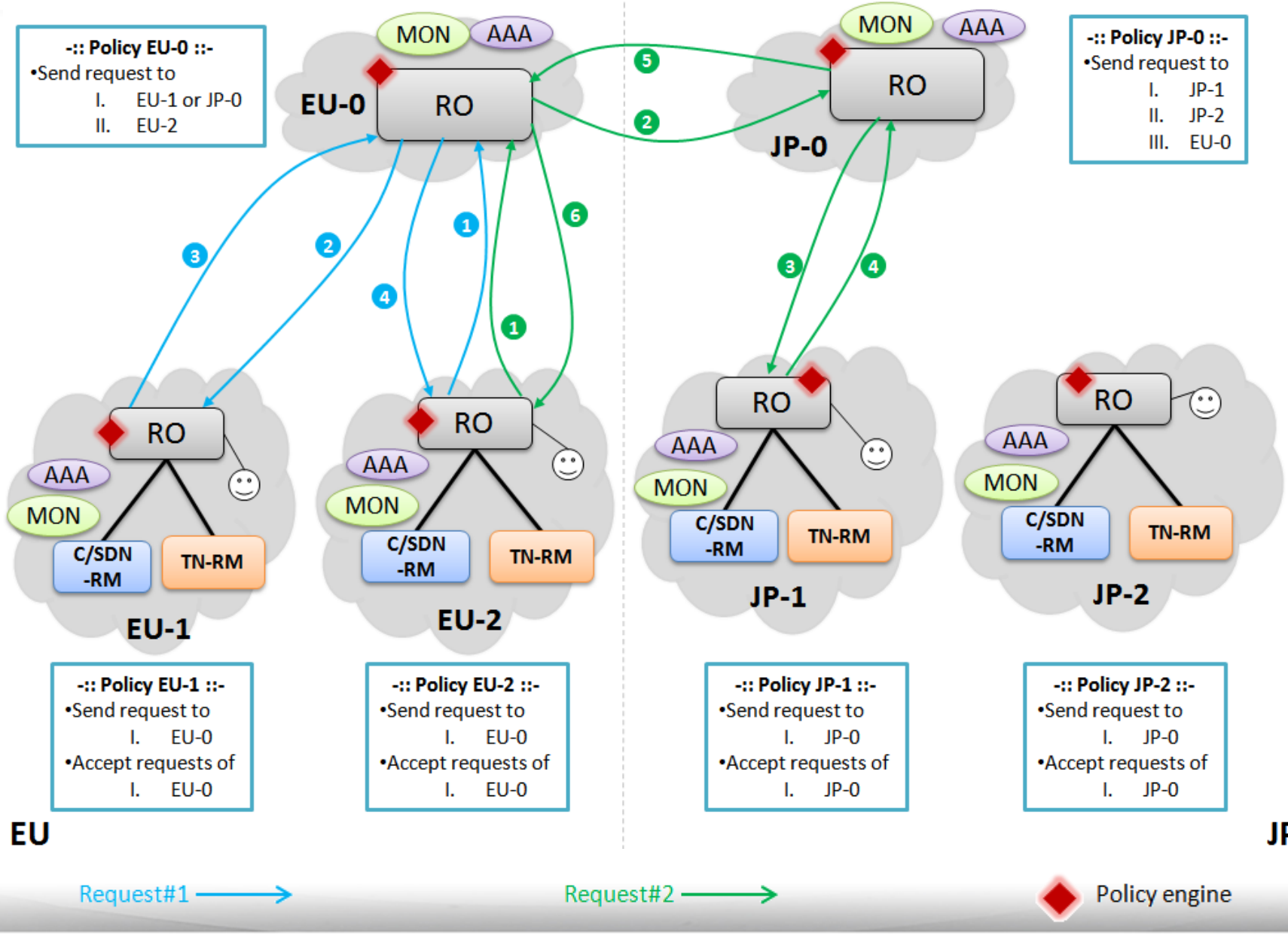
RO Island RO



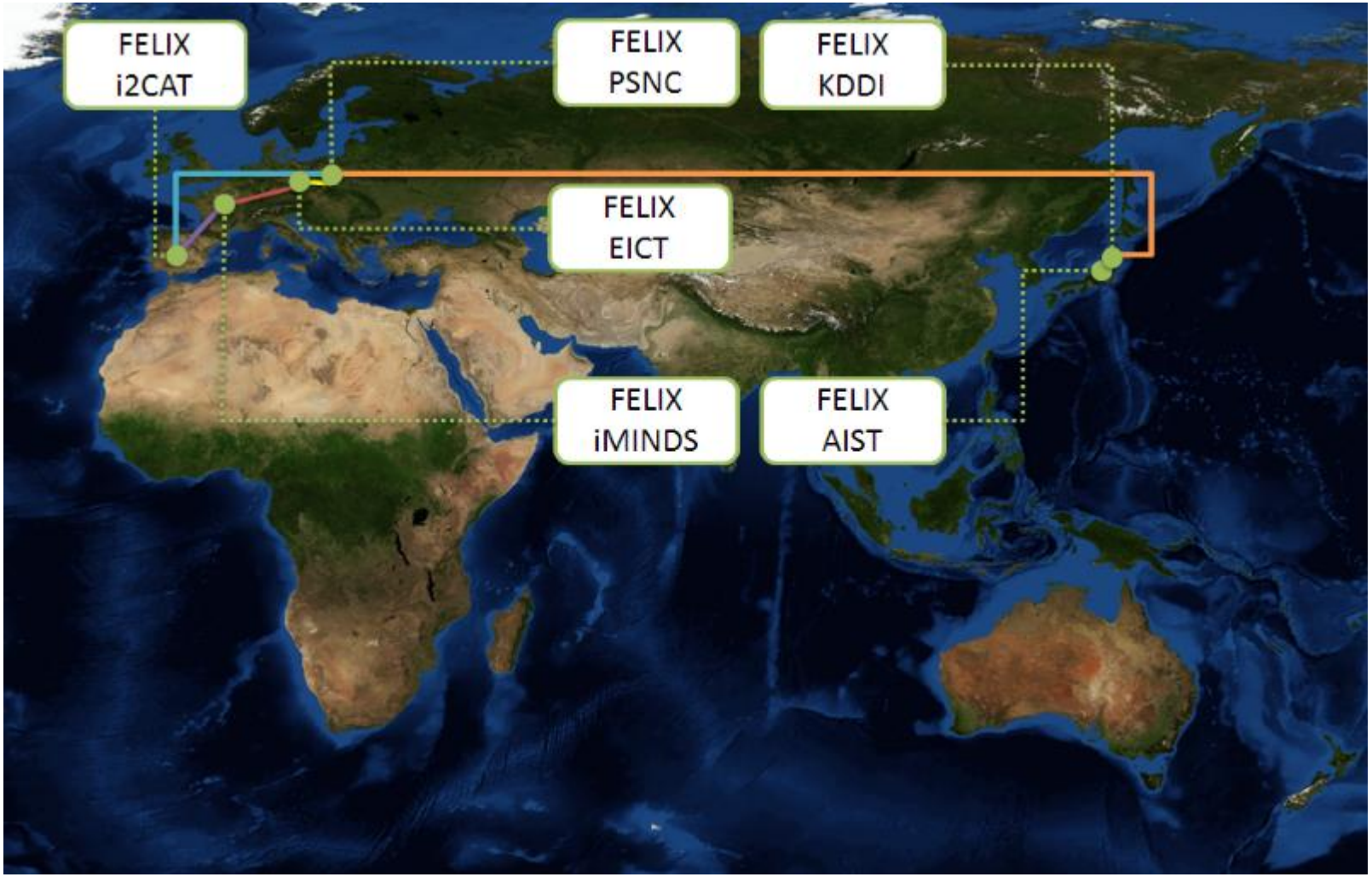
Hybrid

Selected for implementation

FELIX Architecture: Example Request Flows

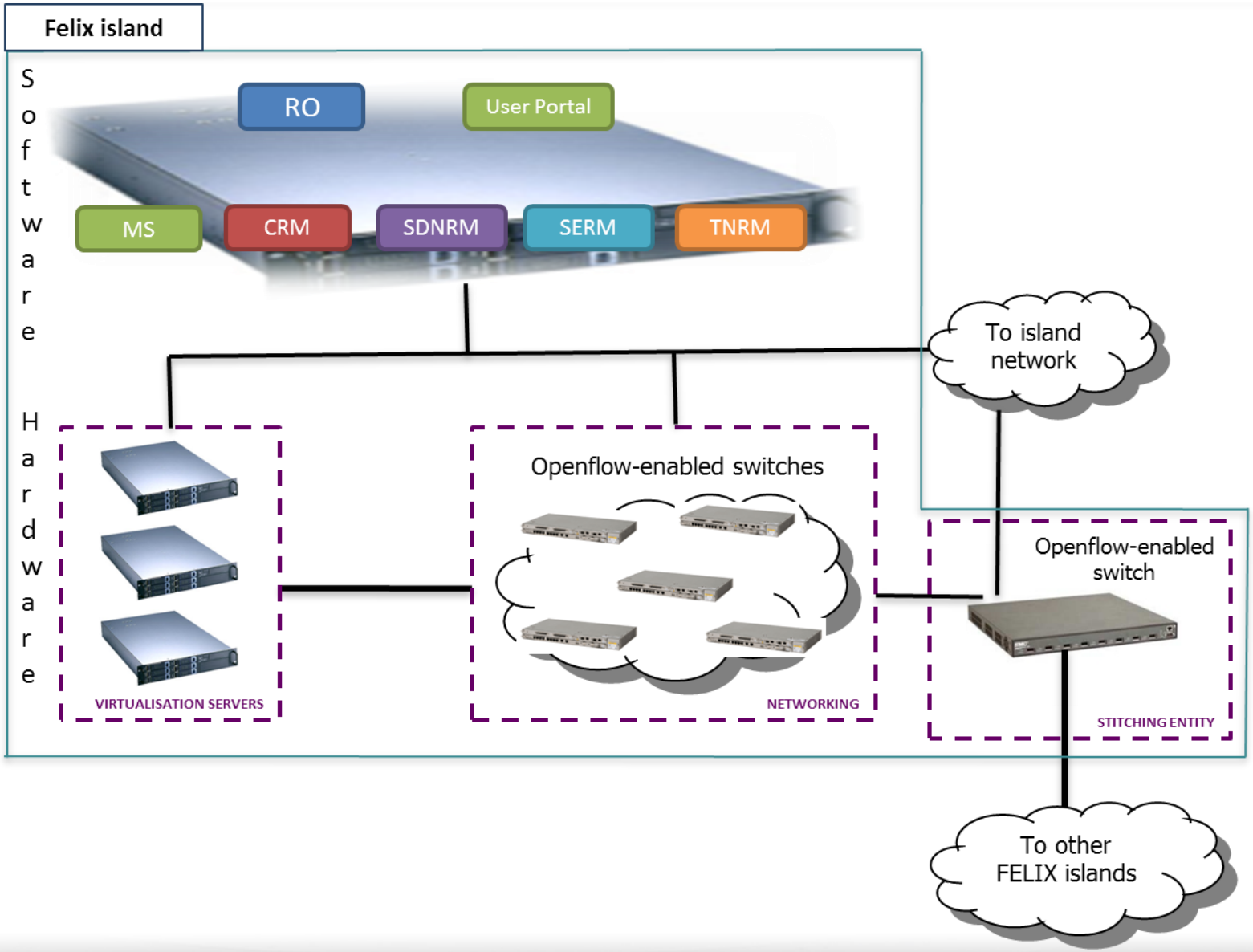


FELIX Experimental Facility

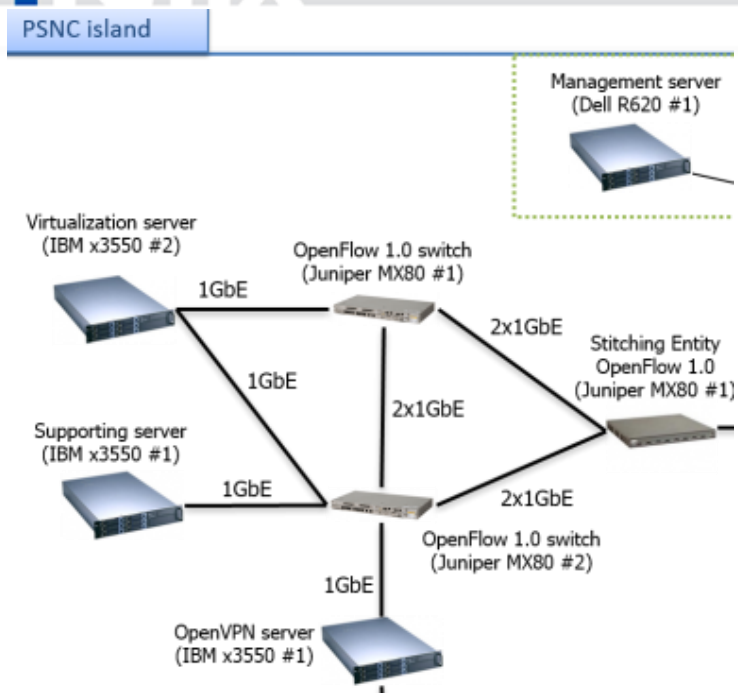


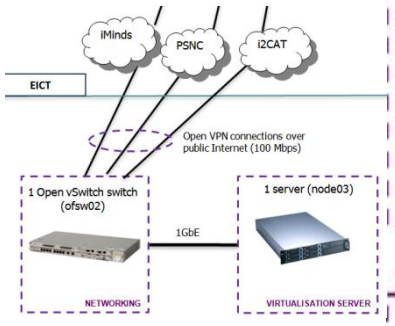
Source: www.ict-felix.eu

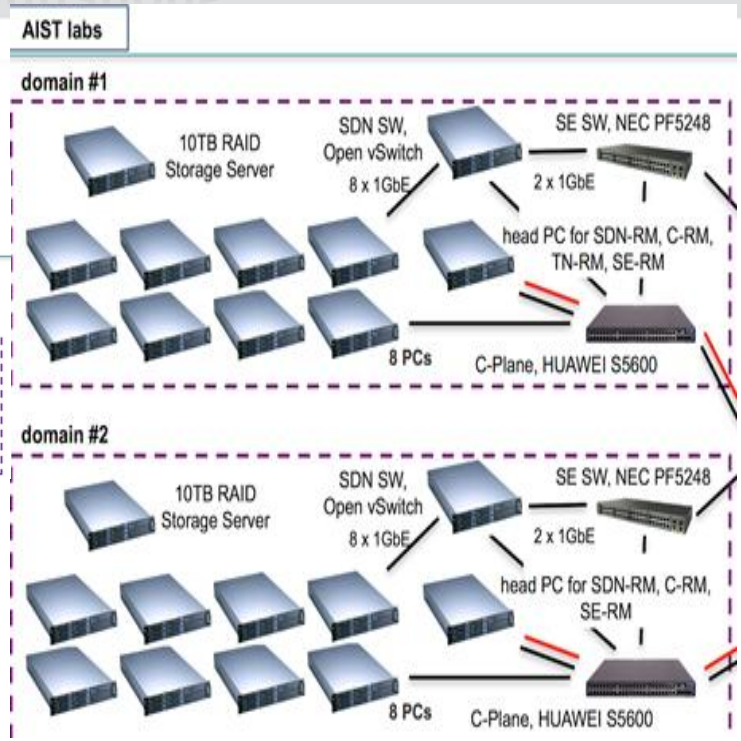
FELIX Island Blueprint

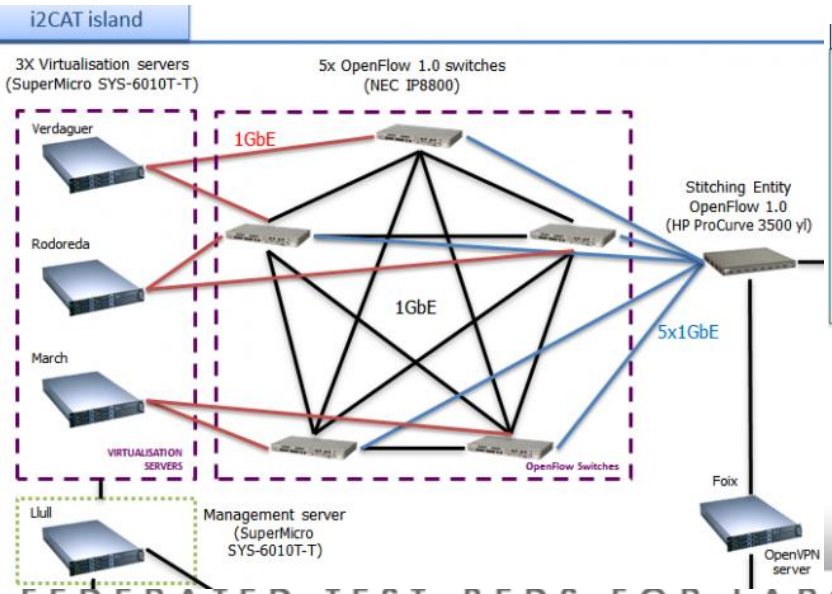


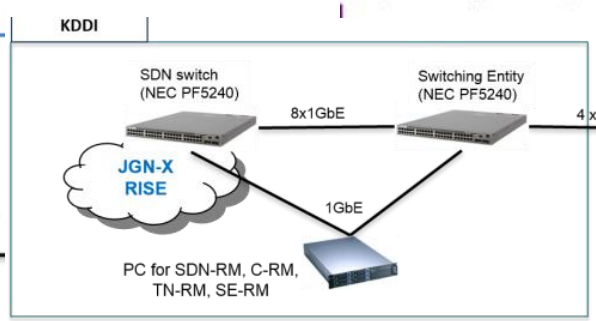
FELIX Island Instantiations

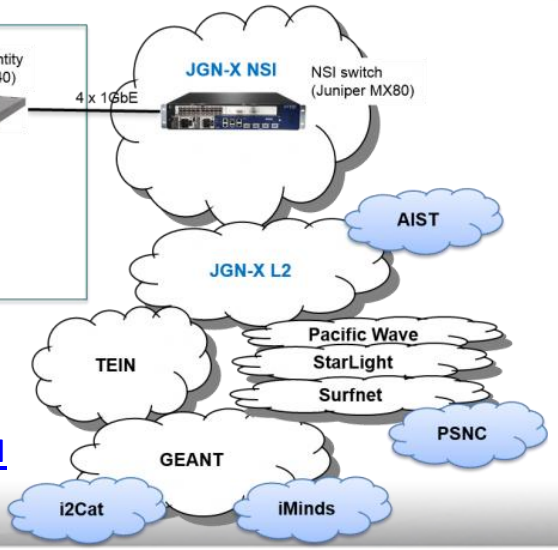












Source: www.ict-felix.eu

Summary

- In software defined infrastructures (SDIs) management and control should be part of the application design
- RFC 7426 provides an excellent starting point for newcomers to SDN regarding
  - Clear terminology and extensive bibliography
  - Understanding of all issues at hand
  - Designing SDI architectures
- RFC 7426 already has a number of “applications”
- SDN Experimental Facilities such as FELIX provide excellent opportunities to think about, design and implement new approaches for SDIs
  - Policy-based hierarchy for resource orchestration
  - FELIX space & User space *vs.* Transport & Service Strata in CLAS
  - Investigate and deploy better management and AAA solutions
  - Reveal limitations of state-of-the-art approaches in practice, esp. for a field with burgeoning interest such as SDN and NFV
- Check out the FELIX deliverables and open source contributions

Bibliography

1. A. Zaalouk and K. Pentikousis, “Network Configuration in OpenFlow Networks”, *Proc. MONAMI*, Wuerzburg, Germany, September 2014, pp. 91-104.
2. E. Haleplidis, et al., “Software Defined Networking (SDN): Layers and Architecture Terminology”, RFC 7426, January 2015.
3. D. Parniewicz, et al., “Design and Implementation of an OpenFlow Hardware Abstraction Layer”, *Proc. ACM SIGCOMM Workshop on Distributed Cloud Computing* (DCC), Chicago, IL, USA, August. 2014, pp. 71-76.
4. M. R. Sama, et al., “Software-Defined Control of the Virtualized Mobile Packet Core”, *IEEE Communications Magazine*, 53(2), February 2015.
5. J. Matias, et al., “Toward an SDN-Enabled NFV Architecture”, *IEEE Communications Magazine*, 53(4), April 2015.
6. G. Carrozzo, et al., “Large-scale SDN experiments in federated environments”, *Proc. SACONET*, Vilanova i la Geltrú, Spain, June 2014, p 1-6.
7. C. Fernandez, et al., “A recursive orchestration and control framework for large-scale, federated SDN experiments: the FELIX architecture and use cases”, *International Journal of Parallel, Emergent and Distributed Systems*. DOI: 10.1080/17445760.2015.1044003

Thanks for your attention!

Questions and comments?

Acknowledgement

Part of this work was conducted within the framework of the FP7 FELIX, which is partially funded by the Commission of the European Union.

Study sponsors had no role in the preparation of this presentation. The views expressed do not necessarily represent the views of the FELIX project, the respective employers, or the Commission of the European Union.