Programmable Management Framework for Evolved SDN

Sławomir Kukliński

Orange Polska and Warsaw University of Technology
Warsaw, Poland

Kraków, May 9, 2014

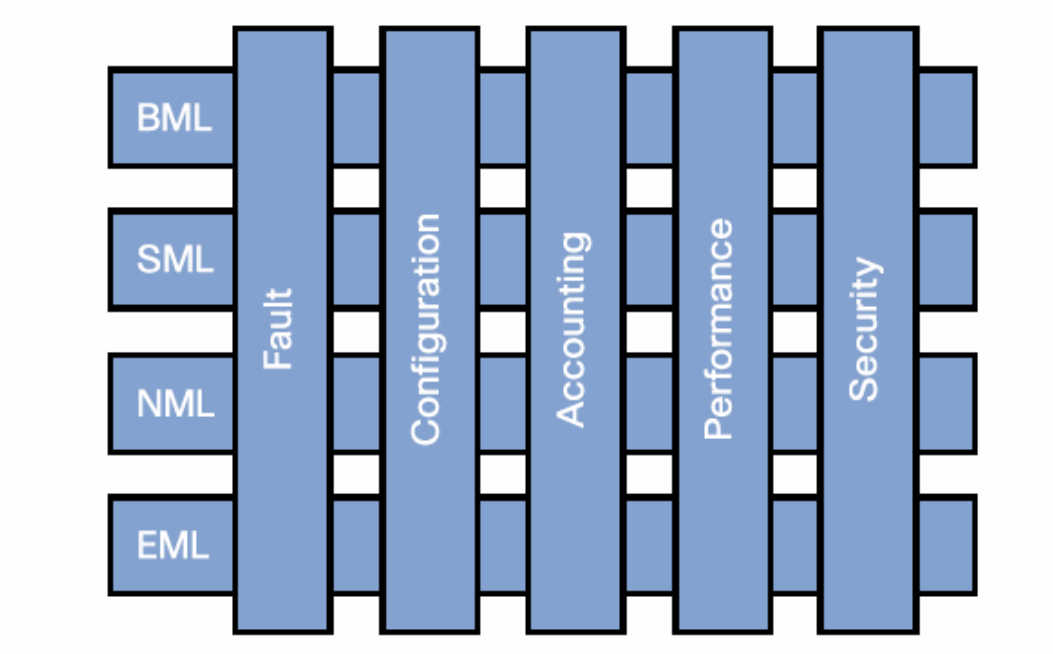
orange™

Problem statement

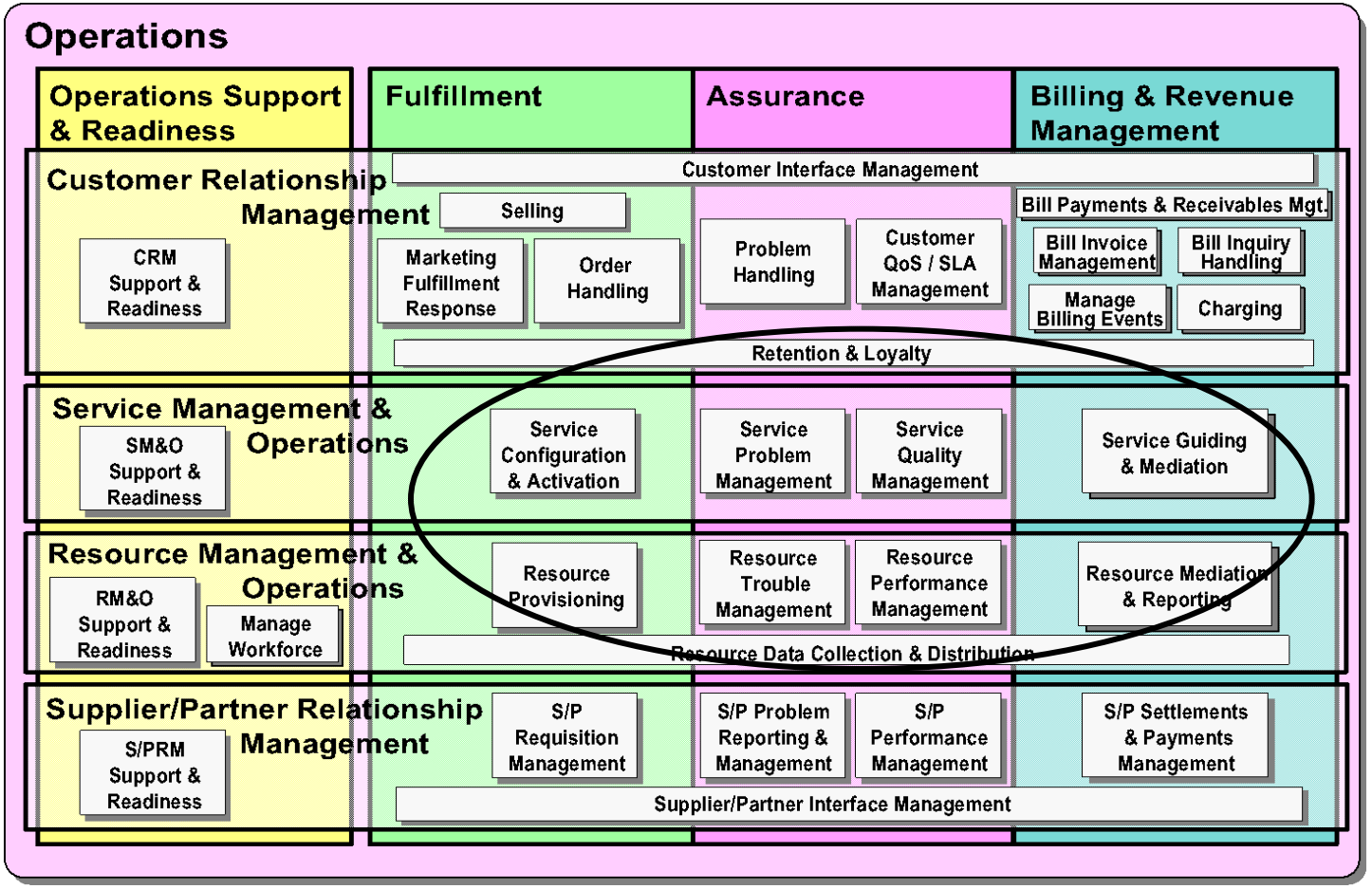
- What (is) should be the scope of SDN management?
- Is management an embedded function' of the controller? If not, do we need special node(s) for SDN management?
- How to split functions between SDN control and management planes? How to provide interaction between them?
- SDN control plane is centralized, should the management plane be centralized as well?
- How to implement management in SDN?

Generic network management

- ITU-T (M.3010/96), ETSI and 3GPP efforts to standardize TMN
- FCAPS Model (ITU-T M.3400/97)

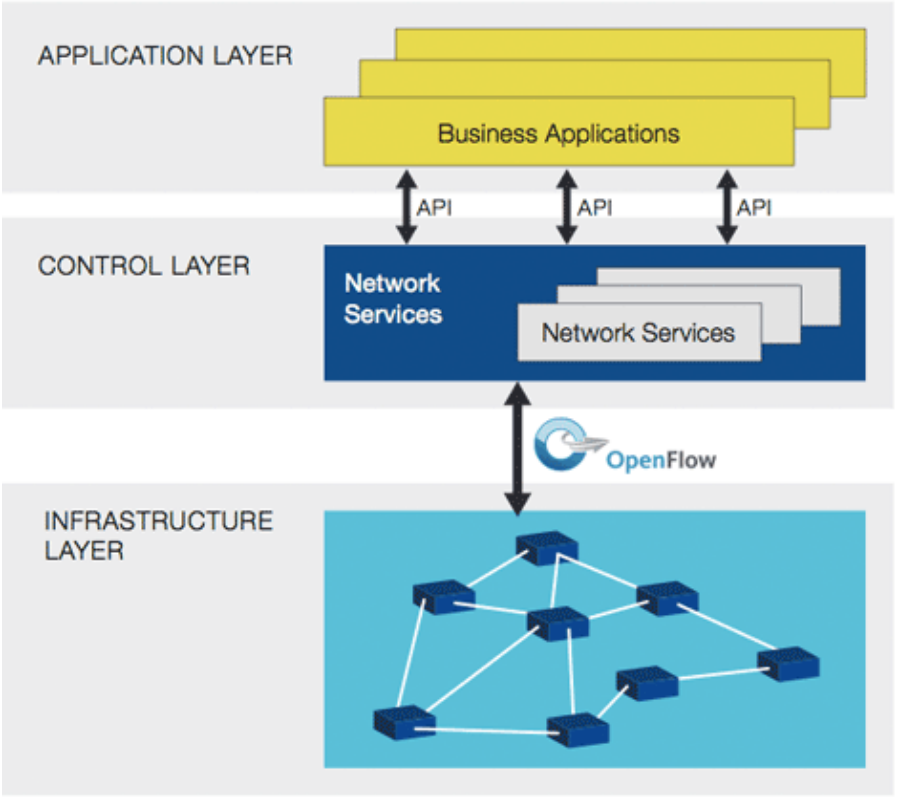


eTOM Operations



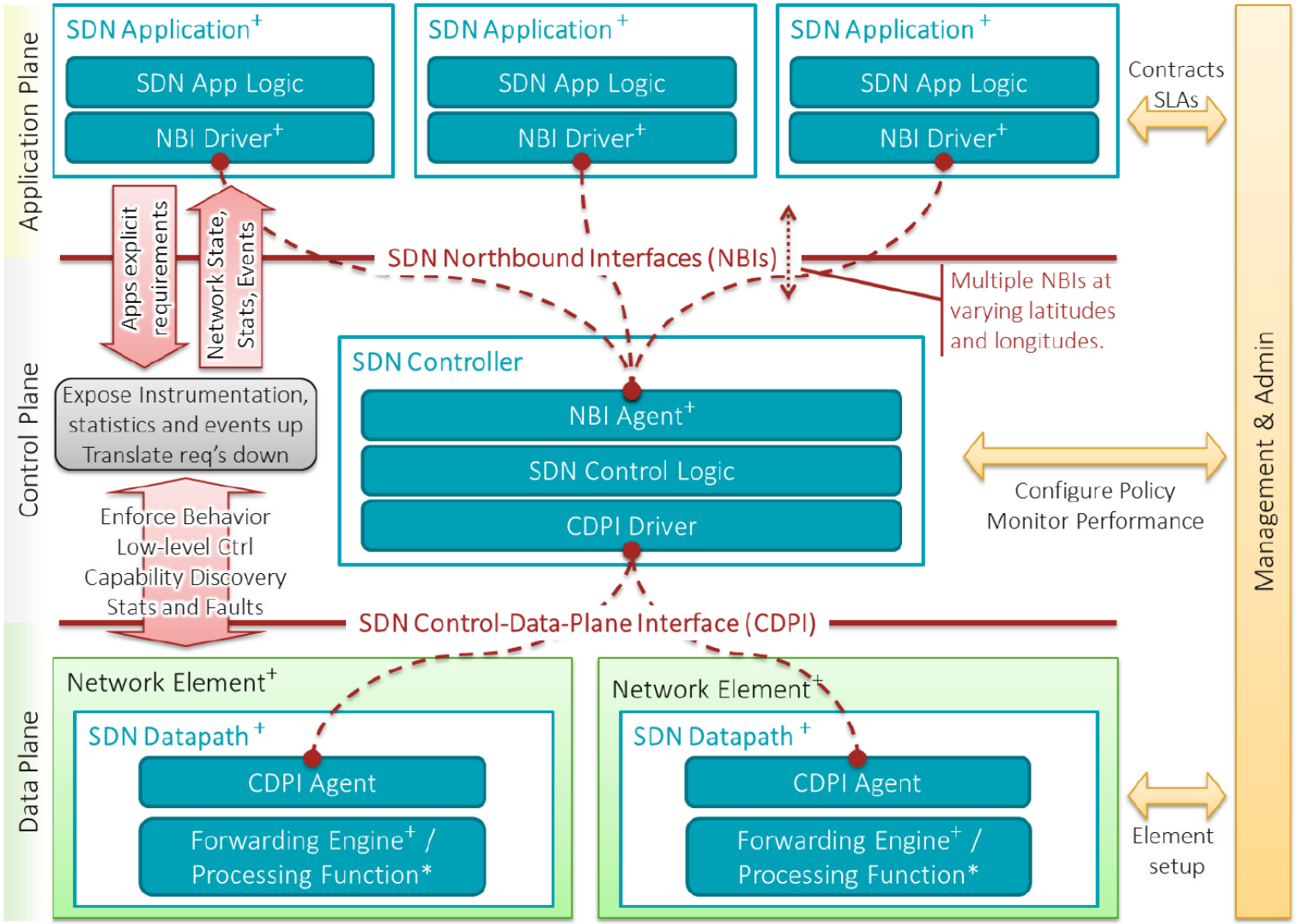
***Source**:* Business Process Framework (eTOM) Enhanced Telecom

SDN original architecture by ONF



Nothing about management - some management operations are supported by the OpenFlow protocol, OF-config added for control plane setup

SDN architecture by ONF (Dec 2013)



+ indicates one or more instances | * indicates zero or more instances

Role of the Management Plane according to ONF

- **Management & Admin:** The Management plane covers static tasks that are better handled outside the application, control and data planes. Examples include business relationship management between provider and client, assigning resources to clients, physical equipment setup, coordinating reachability and credentials among logical and physical entities, configuring bootstrapping. Each business entity has its own management entities. Communication among management entities is beyond the scope of this SDN architecture. One goal of SDN is to subsume many management tasks known from legacy network into the CDPI.

Remark:
There is no clear definition what the managment should do

SDN management functions groups

- SDN hardware related management (Data and Control Planes)
- SDN controller sotware platform (OS) management
- SDN controller ‚functions’ (modules) management, including their life-cycle management (on-fly upgrade)
- SDN Application Plane management (service management)

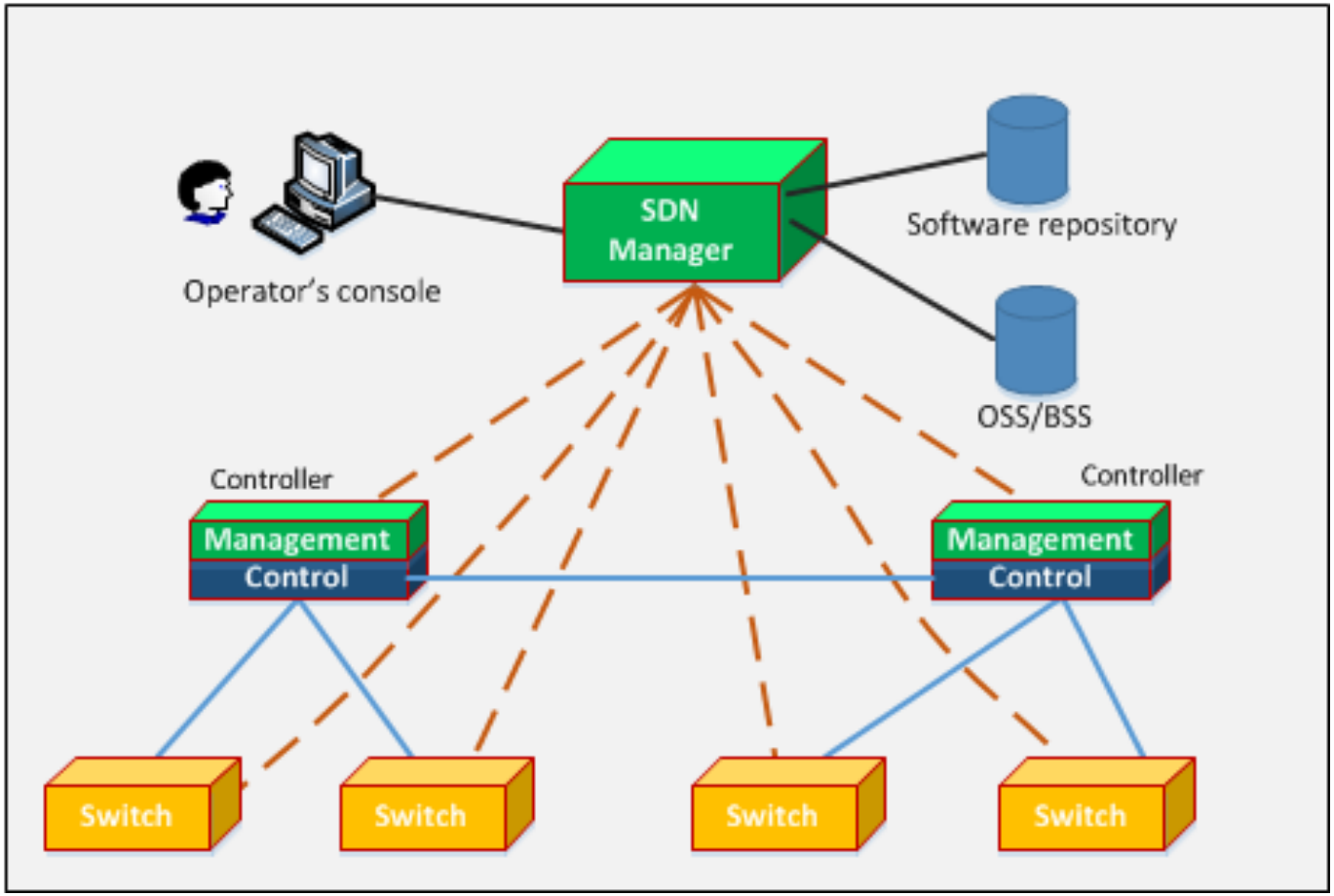
SDN Controller based management

- The concept comes with are some profits: nice integration of the managment with control, simple architecture - a good approach for small networks
- Intelligent management requires significant computing and storage resources, in result the controller complexity increases
- The controller with embedded management will be true single point of failure – in case of failure the management functionalities will be lost
- Security issues in case of the programmability of the controller
- In multi-controller environment we should decide how to split the management functions between the controllers and how the operator should interact with the management system

SDN Manager based management

- Performs most of the management tasks delegating some of them to the controller (if nice cooperation with control plane can be achieved)
- Provides operator console
- Provides interaction with OSS/BSS (billing, CRM)
- Monitors the control plane (DoS attacks, anomalies, control plane congestion) and takes some actions (load balancing of the control plane)

SDN Manager based centralized management



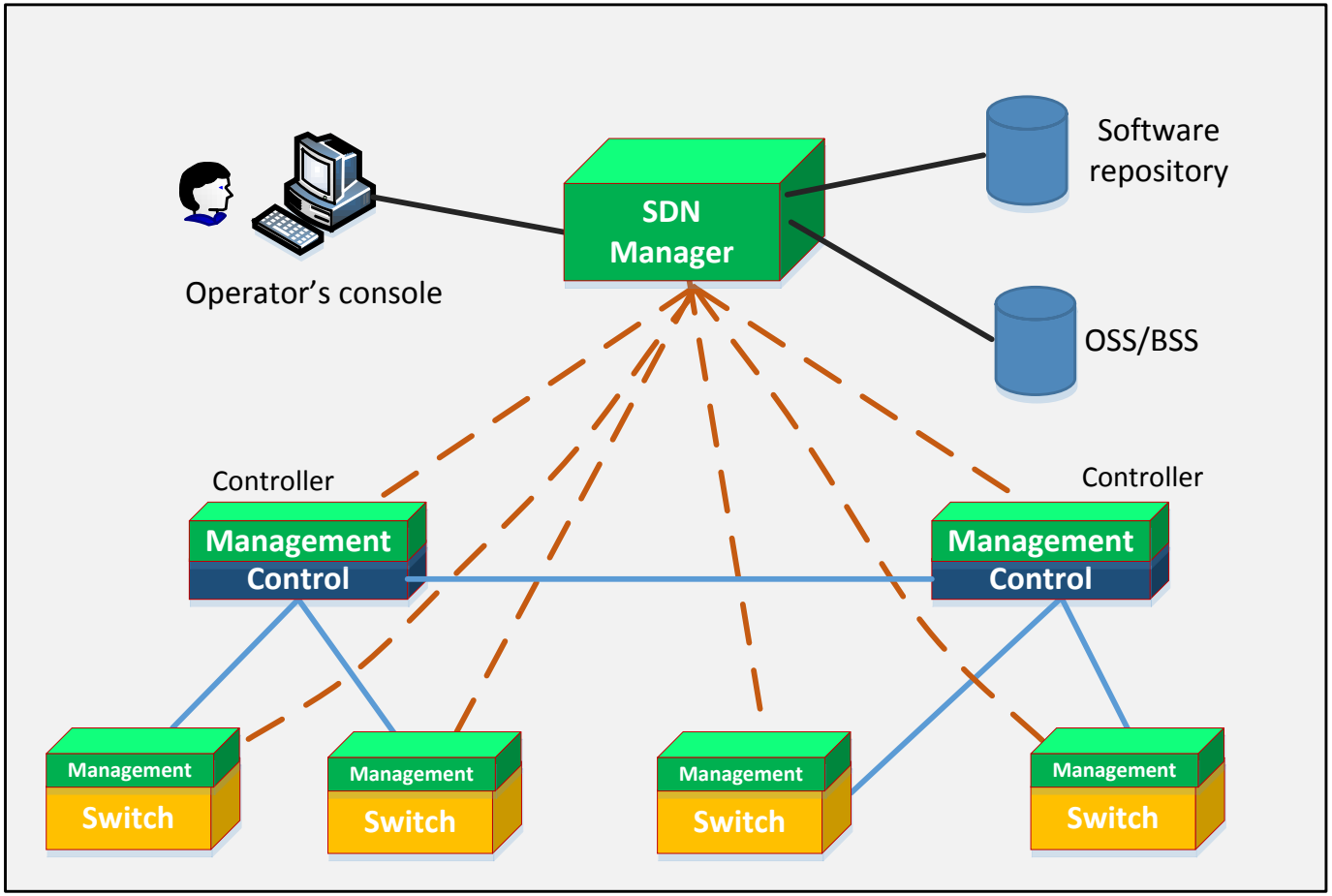
SDN functions split between Controller and Manager

- Manager's responsibility
  - SDN hardware related managment (data and control plane)
  - SDN controller platform managment
  - SDN controller ‚functions' (modules) management (including life-cycle)
  - SDN Application Plane management
- Controller's responsibility
  - Real-time SDN network management

Centralized vs decentralized management

- We like centralization (of the Control Plane) in SDN
  - Centralized traffic engineering, e2e operations
  - Simpler management in comparison to IP networks
  - The switches are simple
- We dislike centralized management
  - Distributed operations allows for local intelligence (plug-and-play nodes)
  - Local actions are faster
  - The monitoring trafiic is minimized
  - The network is more robust
  - The managment is scalable
  - Autonomic Network Management is about distributed network management

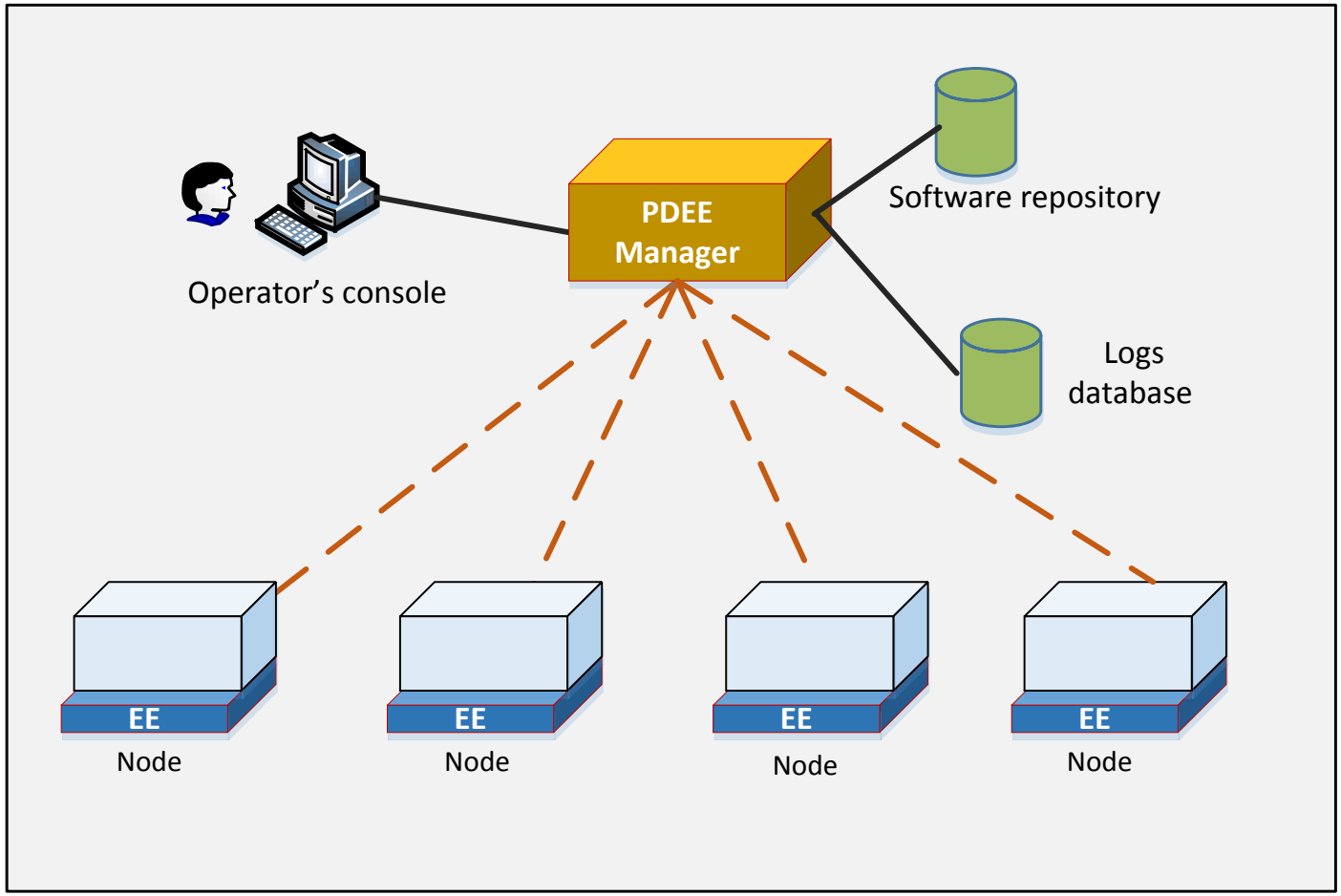
SDN Manager based decentralized management



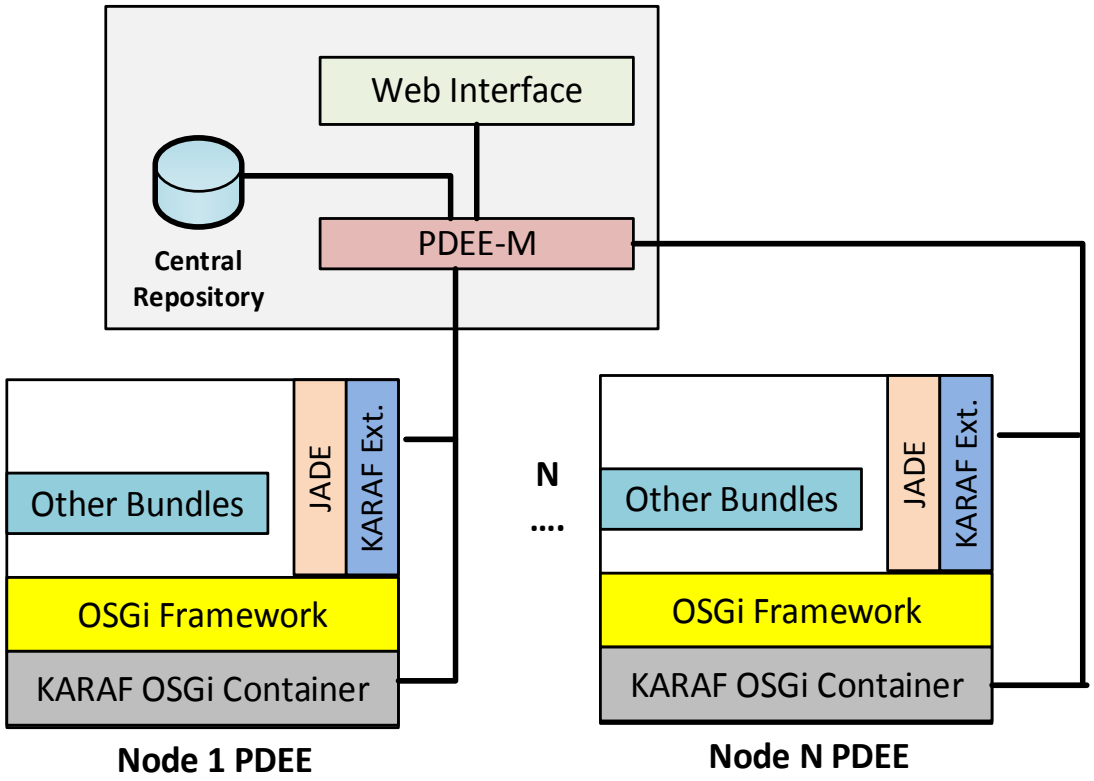
Should the SDN management be programmable?

- The Control Plane is fully programmable
- The Application Plane is programmable
- The Management Plane should be programmable

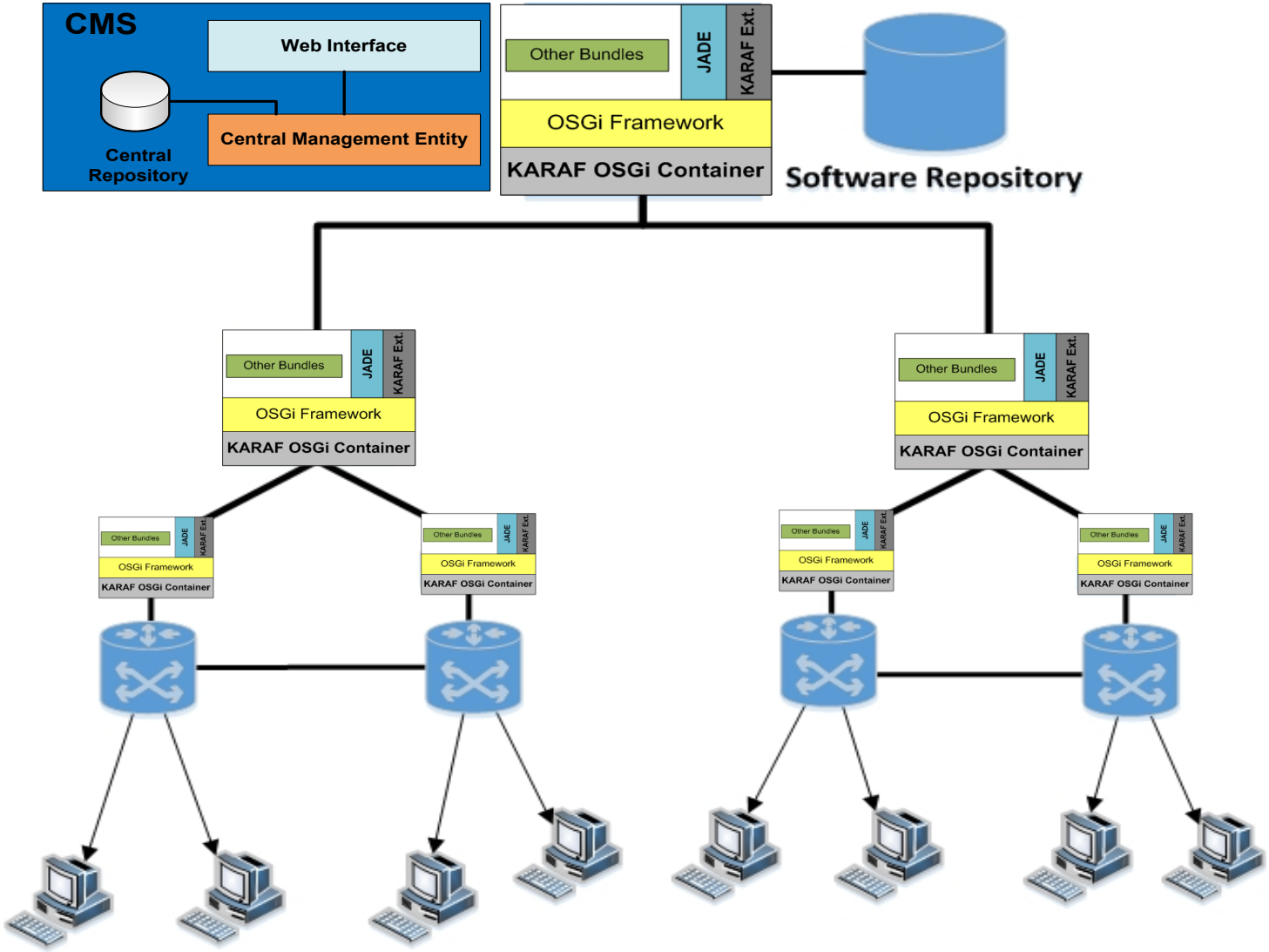
Distributed execution environment based management



PDEE: Programmable Distributed Execution Environment based on OSGi and FIPA



Implementation: PDEE based on OSGi+FIPA



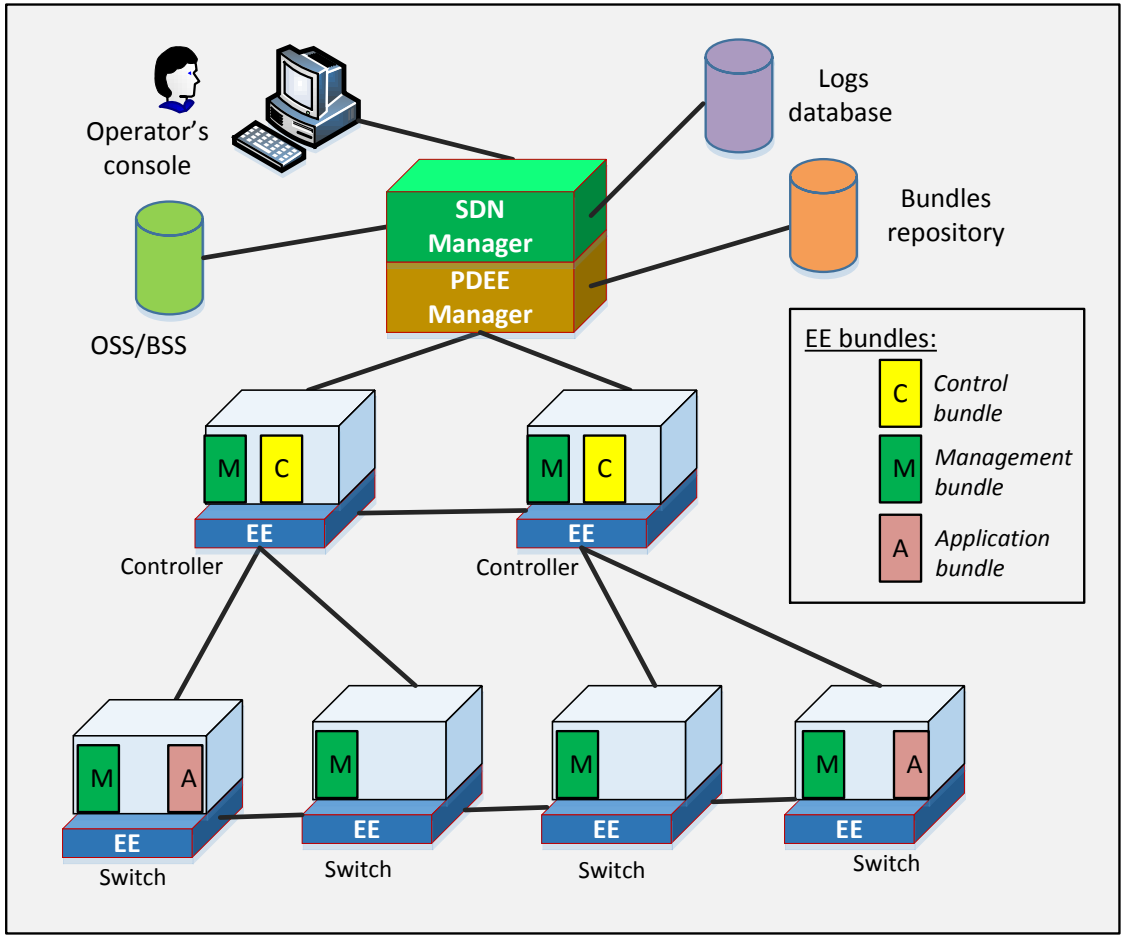
Implementation remarks

- PDEE increases cost and complexity of switches
- Performance can be an issue
- The managment functions of switches should be simple
- PDEE gives an ability to upgrade the control/data plane protocols (CDPI Agent)
- Different switches may have different management functionalities implemented

Impact of PDEE on SDN evolution

- PDEE distributes some management functions, introduces peer-to-peer management information exchange and local intelligence of switches that can be programmed
- At present switch management functions have to interact with existing protocols of switches (OpenFlow, OF-config). PDEE can be used for upgrade or bootstrapping of control-data plane protocols (CDPI Agent)
- PDEE can be used for local implementation of some Control Plane or Application Plane functions (local, edge intelligence)
- PDEE introduces NFV-like SDN with preprogrammed data plane functions (forwarding)

Usage of PDEE for distributed Control, Management and Application Planes



Conclusions

- Management issues in SDN have been discussed
- A need for SDN Manager has been emphasized
- A pragmatic approach to distributed managment in SDN has been proposed. The approach is based on well-established technologies like OSGi and FIPA
- Bootstrapping of OpenFlow versions is supported
- The distributed approach can be used for Control or Application Planes – distributed control in SDN can be introduced
- There is a need for the definition of the low-level protocol to switch fabric (below OpenFlow)
- The implementation of the management functions can be use case dependent, there is no need to define them a priori

Thank you!

23