Software-Defined Networking

Daphné Tuncer
Department of Computing
Imperial College London (UK)
dtuncer@ic.ac.uk

Agenda

Part I: Principles of Software-Defined Networking (SDN)

1. Why a lecture on SDN?
2. What is SDN?
3. Architecture and main functions

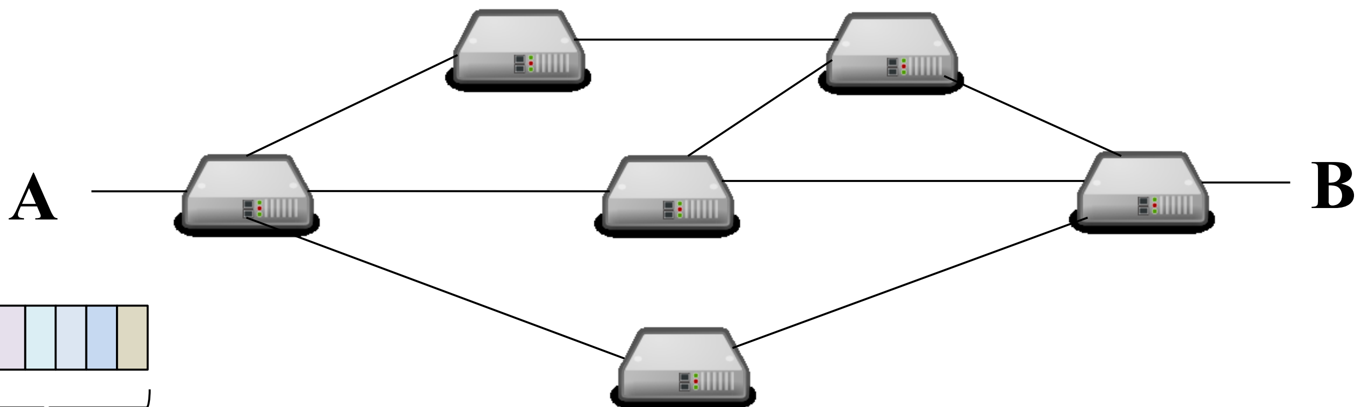
Part II: OpenFlow

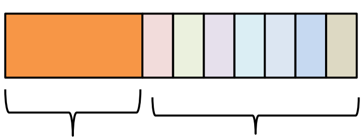
1. Protocol and southbound interface
2. Packet forwarding in OpenFlow-enabled networks
3. OpenFlow controllers
4. Short demo using Mininet

Quick Recap on Networking

Basic networking principle: enabling communication between end points

1100100011
0011111010
1111001110
1010111000



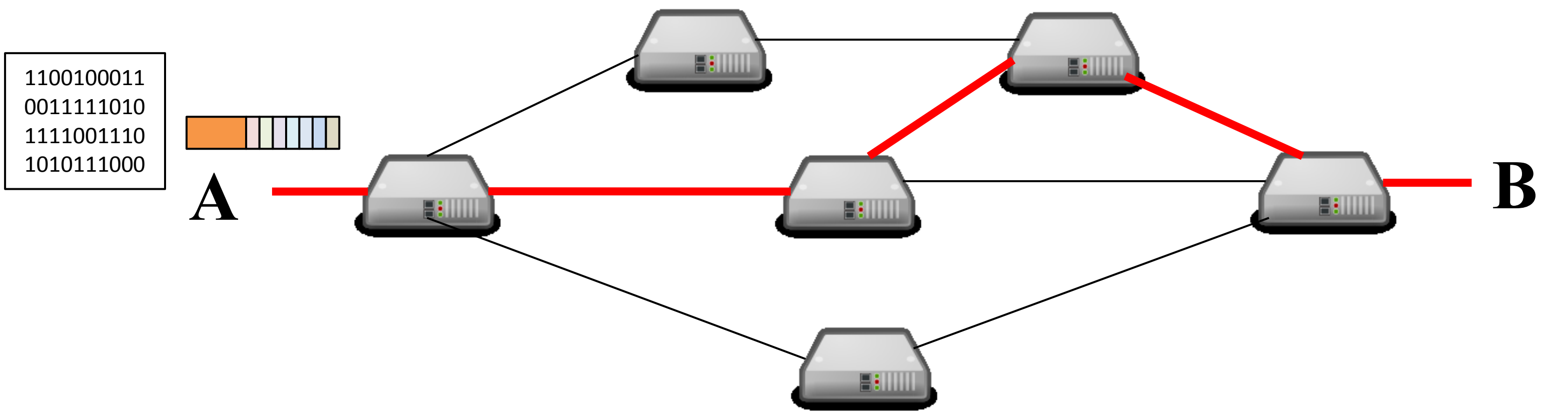


- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

OSI model

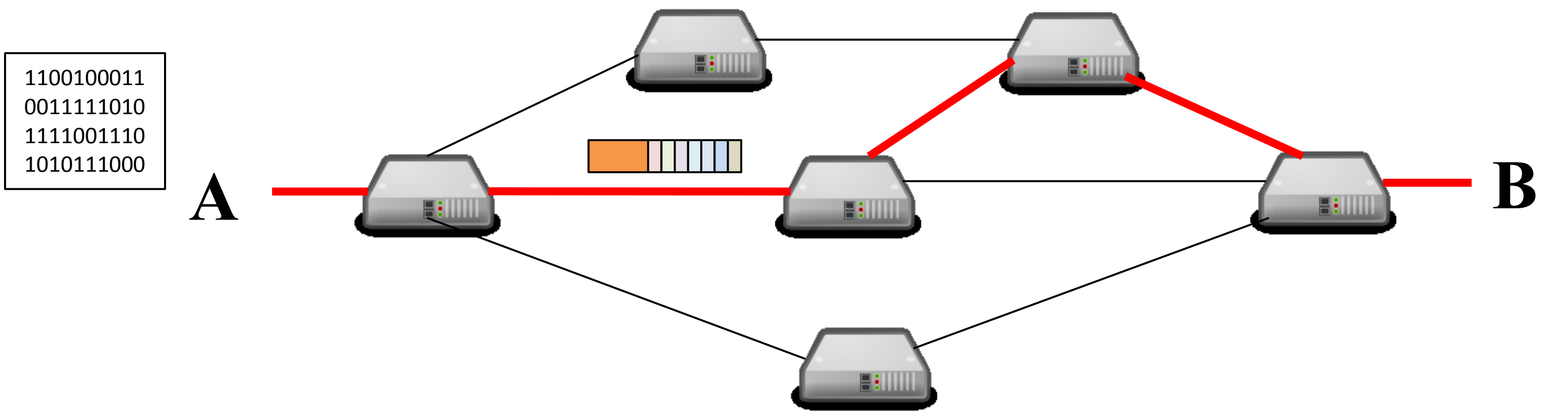
Quick Recap on Networking

Basic networking principle: enabling communication between end points



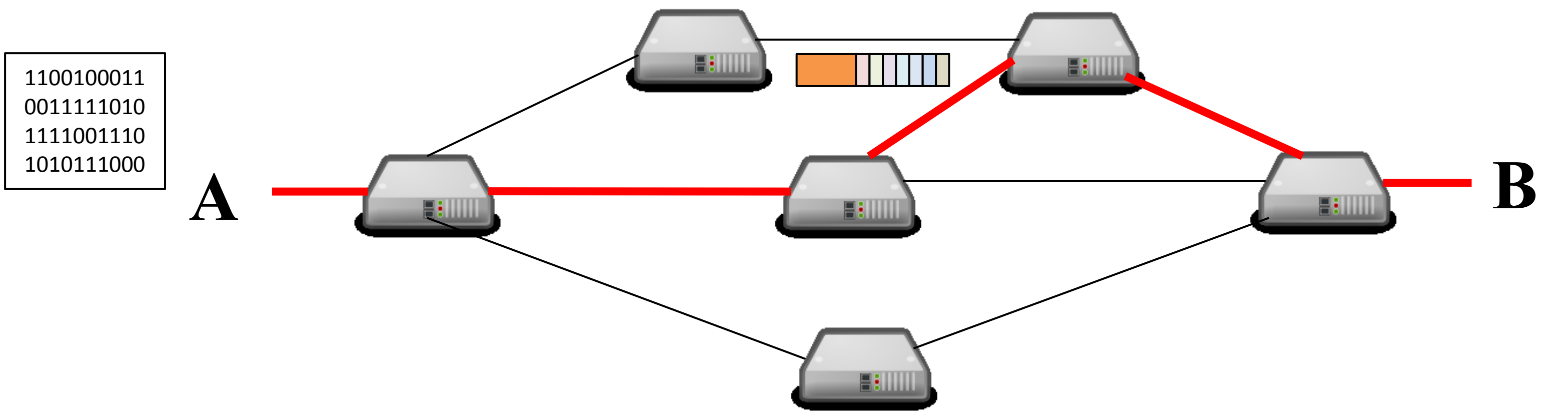
Quick Recap on Networking

Basic networking principle: enabling communication between end points



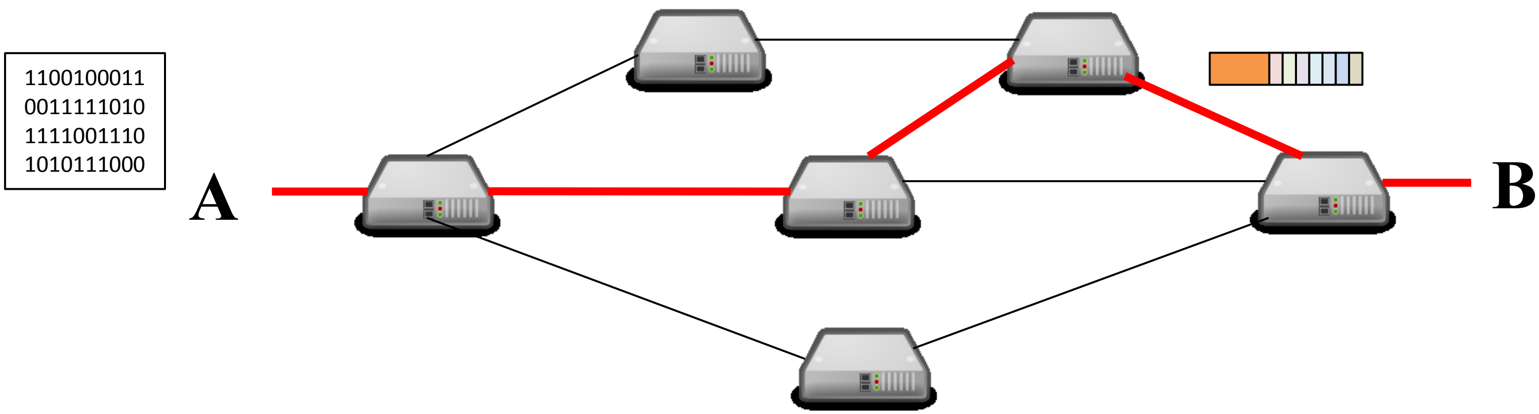
Quick Recap on Networking

Basic networking principle: enabling communication between end points



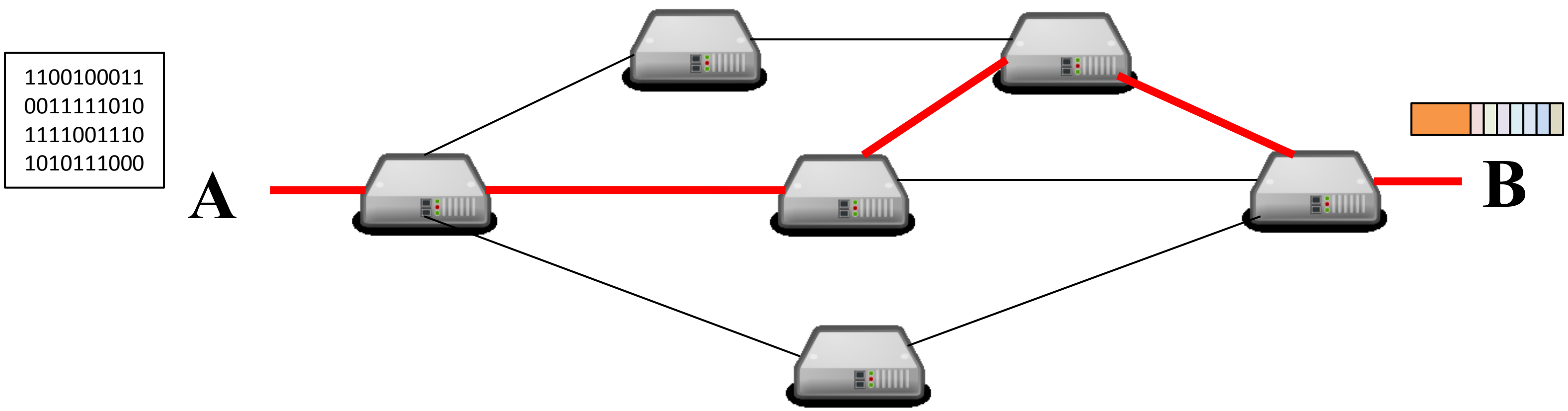
Quick Recap on Networking

Basic networking principle: enabling communication between end points

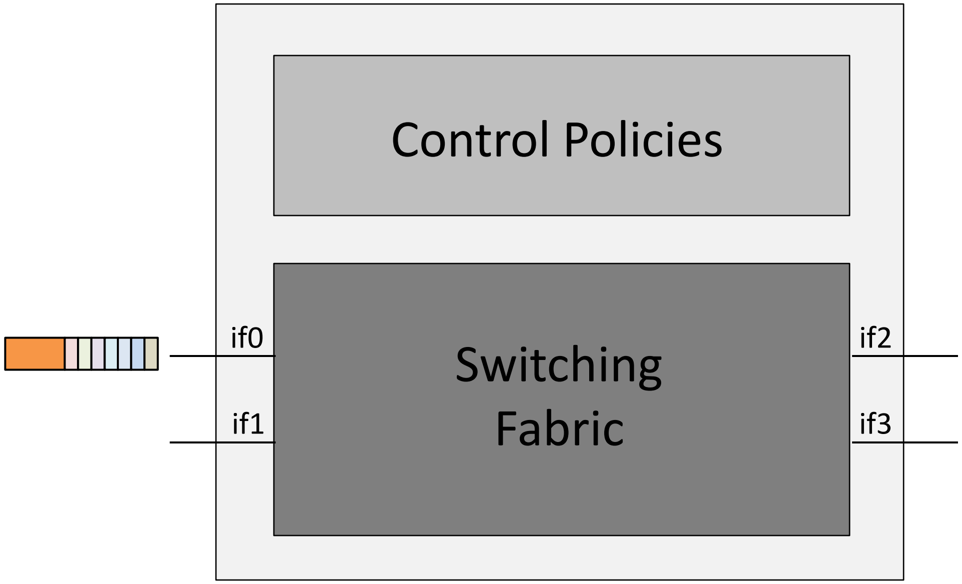


Quick Recap on Networking

Basic networking principle: enabling communication between end points



Zoom on the Network Device



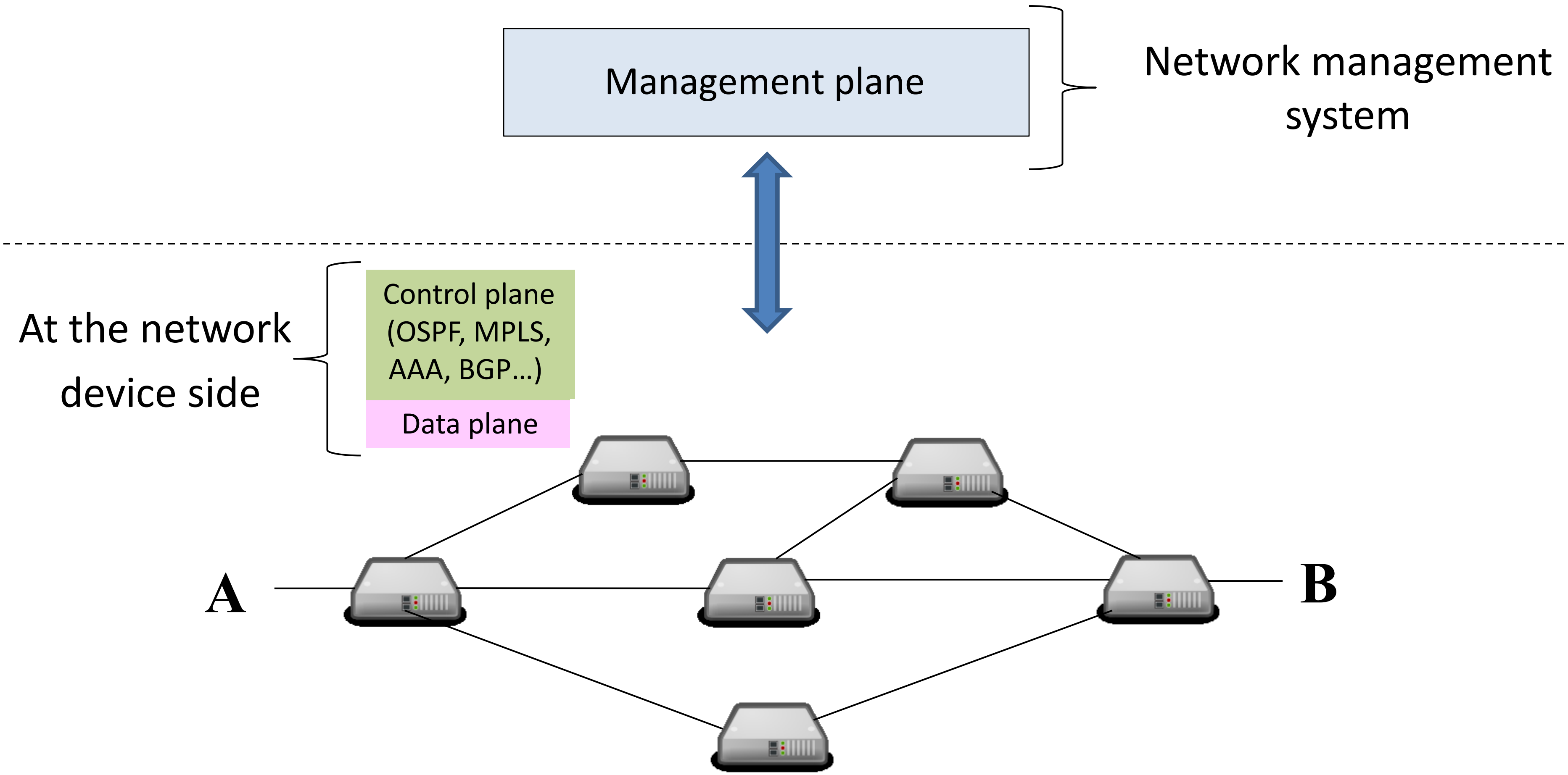
To control the forwarding behaviour of packets

To process and deliver in-coming packets

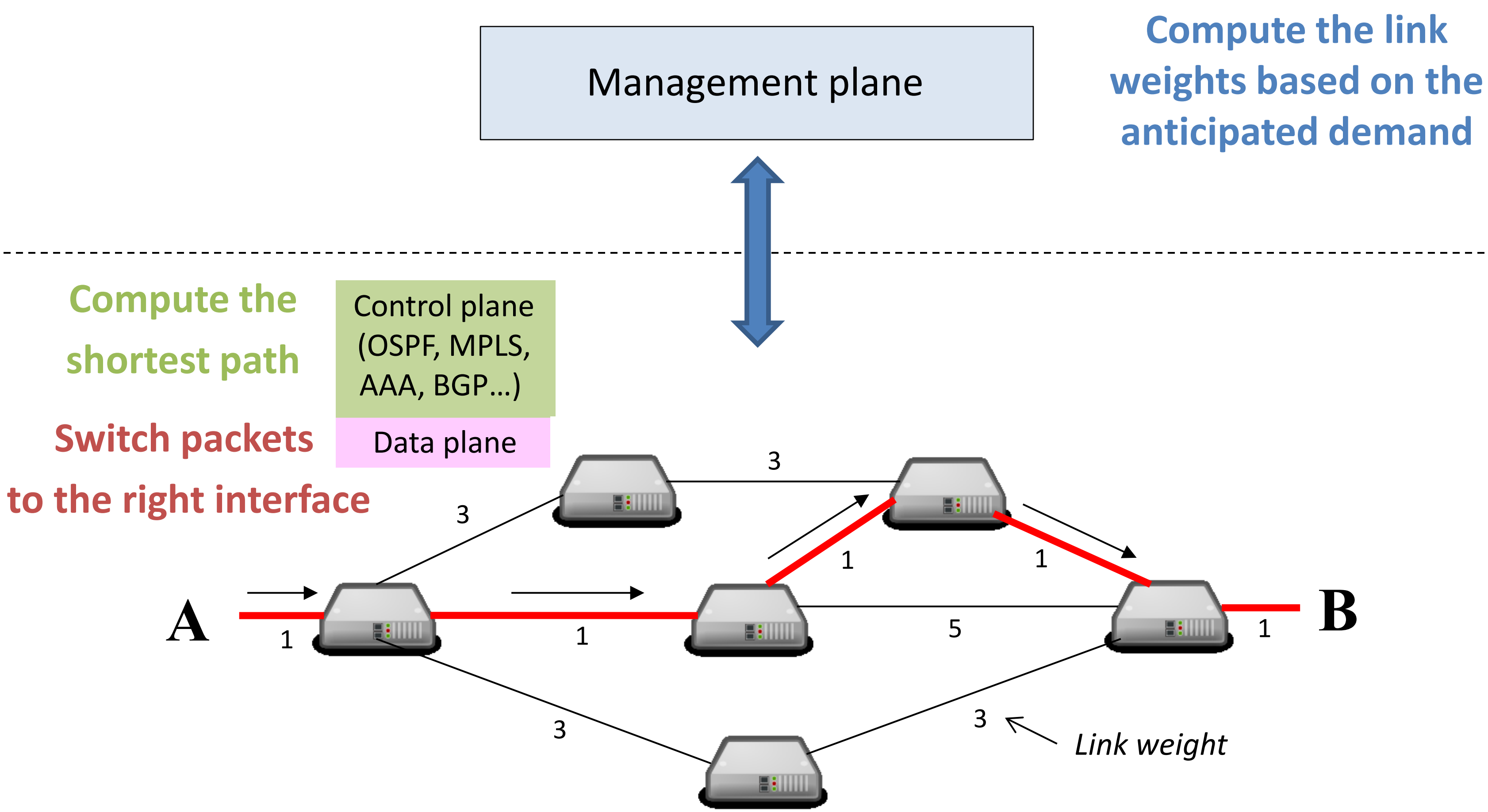
The Three Networking Planes

- Data Plane
  - Processing and delivery of data packets by a network device
  - Data plane operations always target data packets
  - Examples: packet forwarding, marking and dropping
- Control Plane
  - Control on how/where data packets are processed/delivered
  - Defines the policies driving how to handle the in-coming data packets based on network protocols (MPLS, OSPF, BGP, AAA, *etc.*)
  - Often involves local computing tasks (*e.g.,* shortest path computation in IP routing) as well as packet exchanges for routing and signalling
- Management Plane
  - Monitoring and configuration of the network

The Three Planes in Traditional Networking



Illustration



Some observations

- The traditional networking model follows a design where software for the control plane cannot be separated from the packet processing hardware in the data plane.
  - Vertical integration of the control plane and date plane in a single box
- It is based on complex, closed and proprietary vendor-specific implementation (CISCO, JUNIPER, HUAWEI, *etc.*)

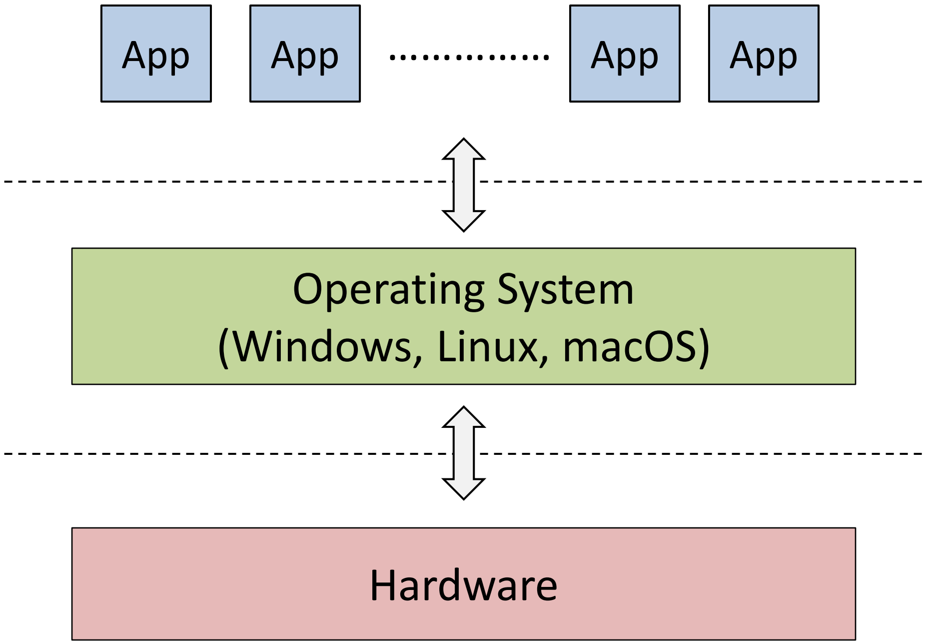
Limitations of the Traditional Model

- Managing the infrastructure is hard: configuration scripts are vendor-dependent, no standardised model

- Upgrading network protocols or deploying new ones is difficult: significantly impairs innovation and possibility for the system to evolve

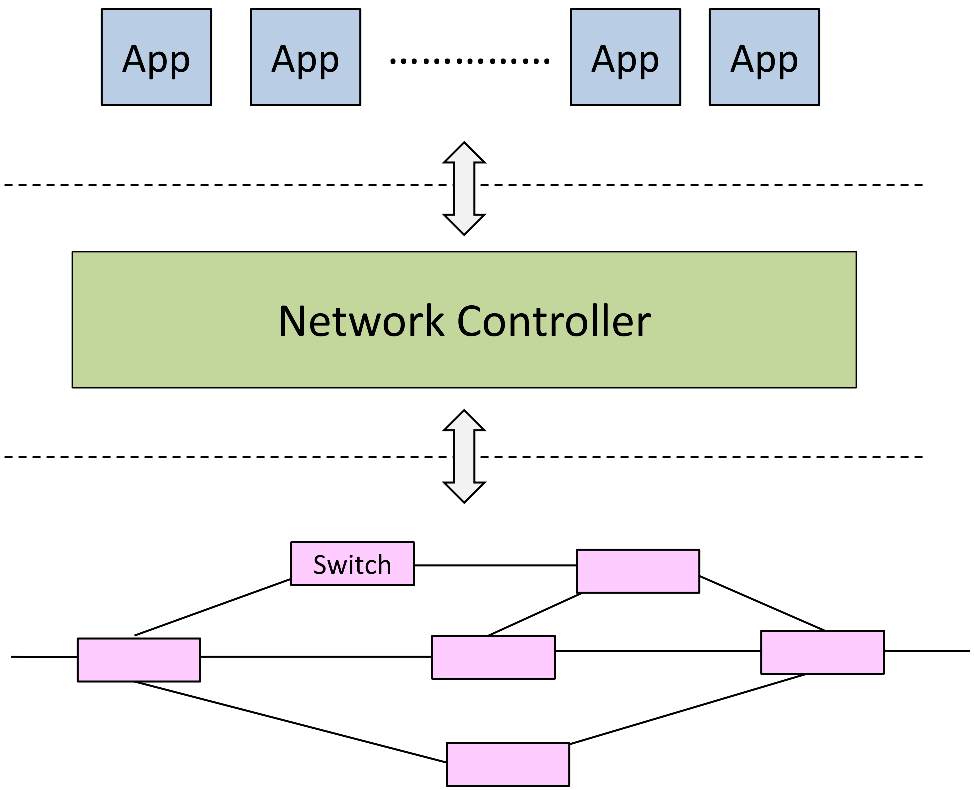
$\Rightarrow$ It had become necessary to re-think this traditional model

Today's Computing System



- Layered architecture and modular structure
- Builds upon well-defined abstractions hiding the details of the underlying layers
- Communication between layers enabled through open interfaces

Model Transposition to the Network



Management applications

Network operating system

Network hardware

Software-defined Networking

- Networking design enabling the separation between the control plane and the data plane

In practise:

- Control functions moved outside the device boxes and placed at the dedicated controller entity (Network Operating System - NOS)
- Device boxes left ‘dummy’ (do simple packet processing)
- Advanced network intelligence flexibly added to the controller only
  - ➢ The data plane is untouched
- Flexible open interface using global network abstractions

Multiple Benefits

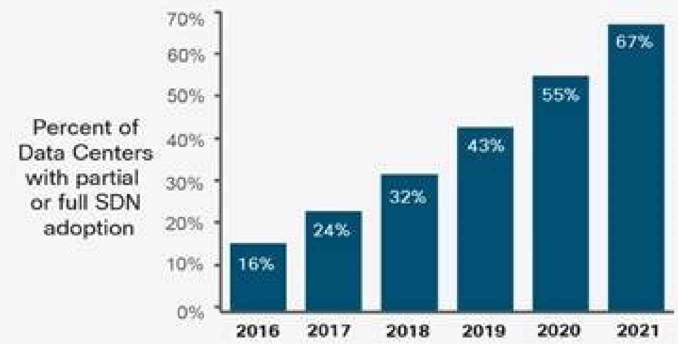
- Programmability through *software-defined* network control functions
  - Network operators can ‘programme’ the network by adding control policies to the controller
  - Multiple network control functions can coexist at the controller side
- Fast pace evolution of networking technologies
  - It is not necessary to follow hardware upgrade to deploy newly standardised control plane networking technologies
- Cost reduction
  - Simpler network devices in the data plane
- Enabling innovation
  - Novel networking techniques can be easily tested and reconfigured at the controller side

SDN in the Wild

Open Networking Foundation

Strong penetration in the datacenter world

Google, Microsoft, Facebook, *etc.*



Source: Cisco Global Cloud Index, 2016-2021.

Telecom Operators

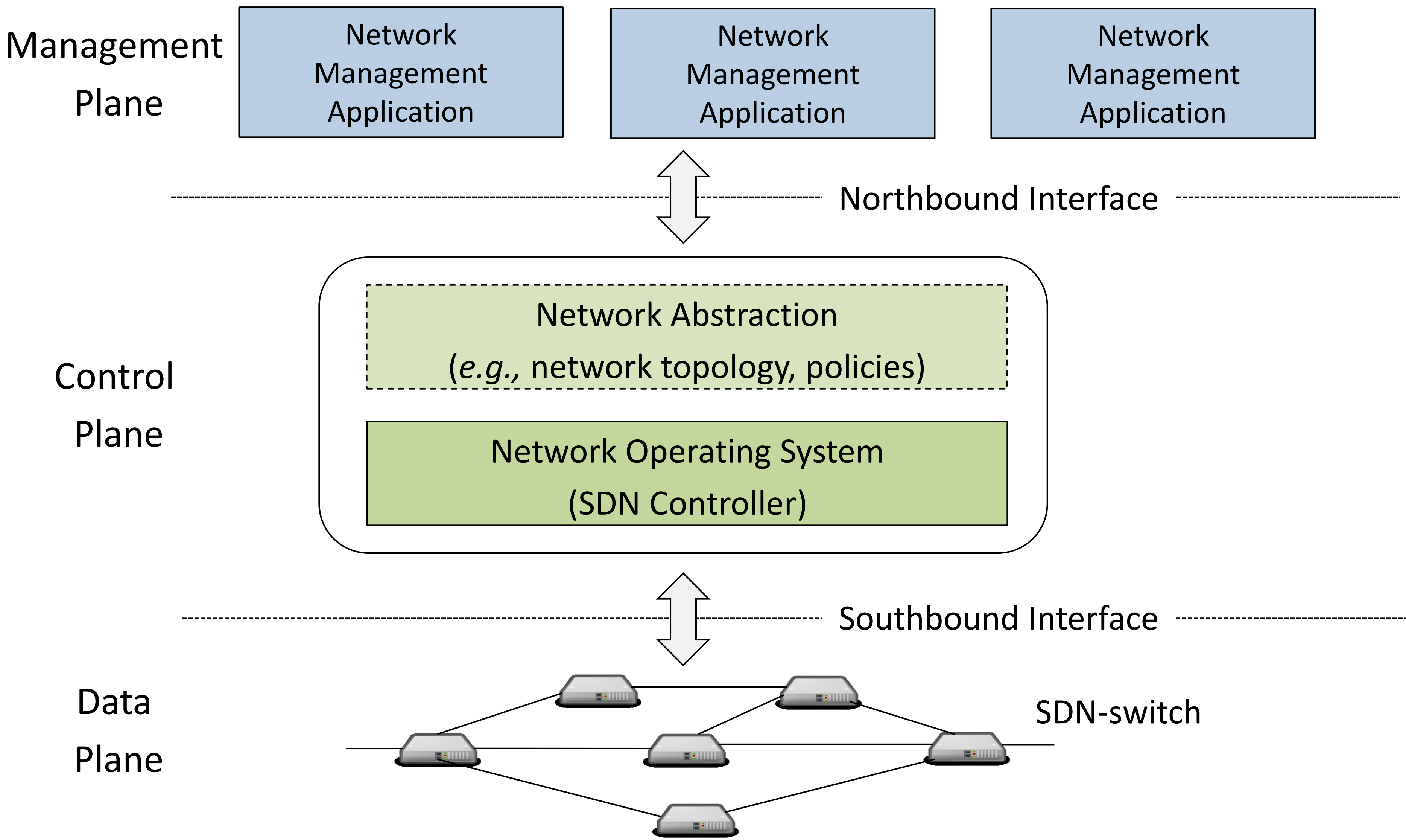
NTT, AT&T, Deutsch Telecom, *etc.*

Academic research

Vendors

IBM, NEC, Nicira, Cisco, Huawei, *etc.*

SDN Architecture



SDN Interfaces

- Northbound Interface (NBI)
  - The interface supporting the interaction between the controller and external network management applications
  - Fulfilment of requirements and policies from applications
  - Provides network abstractions to the applications

- Southbound Interface (SBI)
  - The interface supporting the interaction between the controller and individual SDN-enabled switches in the data plane
  - Enforcing different packet treatment rules for each flow
  - Typically supported by the OpenFlow protocol (next lecture)

Comparison Traditional Network *vs.* SDN

Traditional model	SDN model
Vertical integration between the packet processing hardware and the control software in the same box	The controller does not need to reside in the same box as the packet processing hardware
Network control functions operate in a distributed manner, no global view	Network control functions can operate on a global view of the resources
Network abstraction (*e.g.,* topology) embedded in the distributed network control algorithms	New network abstractions in the control plane (*e.g.,* graph, database)

Some Open Research Challenges

- Abstraction and interfaces
  - Programming language for specifying network functions, *i.e.,* control policies
  - Implementation of a northbound interface: which abstraction to expose to the applications?
- Performance in terms of scalability
  - How many controllers?
  - The communication bandwidth between controller and devices is a scare resource
- Security
  - Open communication channel between the controller and the devices
- Is the southbound interface really standardised?

L. Hendriks, et al., "Assessing the quality of flow measurements from OpenFlow devices.,“ in Proc. of the *8th International Workshop on Traffic Monitoring and Analysis (TMA)*, 2016.

Suggested Reading

[1] H. Kim and N. Feamster, "Improving network management with software defined networking," in *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114-119, February 2013.

[2] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: an intellectual history of programmable networks," in *SIGCOMM Comput. Commun. Rev.* 44, 2, 87-98, April 2014.

[3] T. Koponen, et al., "Onix: a distributed control platform for large-scale production networks, " In *Proc. of the 9th USENIX conference on Operating systems design and implementation* (OSDI), pp. 351-364, 2010.

[4] M. Casado, N. Foster, and A. Guha, "Abstractions for software-defined networks," Communications of ACM, 57, 10, pp. 86-95, September 2014.

[5] S. Clayman and D. Tuncer, "Lessons from Operating Systems for Layering and Abstractions in 5G Networks," in the Proc. of the 3rd IFIP/IEEE International Workshop on Management of 5G Networks (5GMan), Taipei, Taiwan, April 2018.

Contact

Dr. Daphné Tuncer

Department of Computing
Imperial College London
United-Kingdom

dtuncer@ic.ac.uk

http://www.doc.ic.ac.uk/~dtuncer/