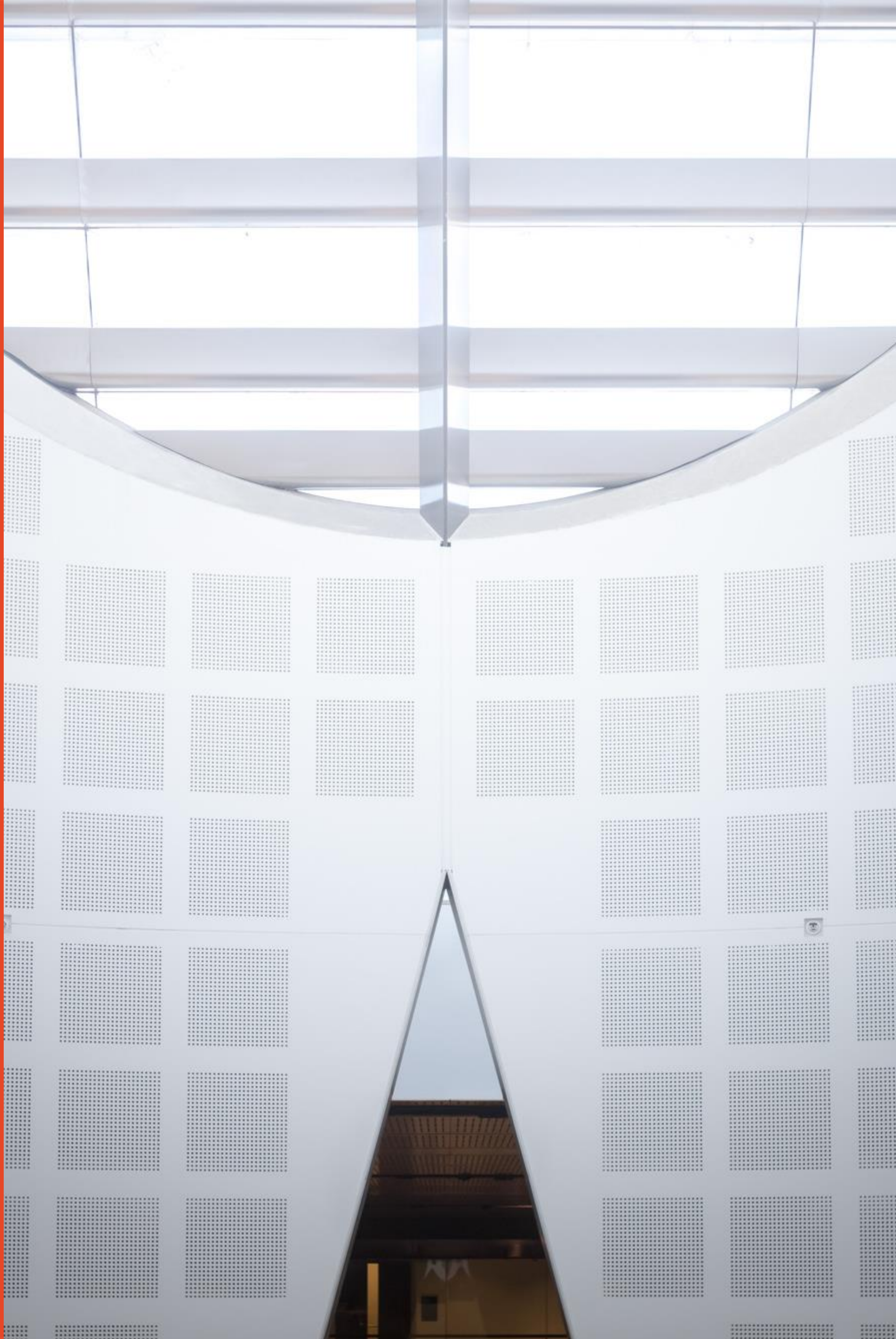
# Topic 4: OpenFlow Switch & SDN Architecture

**Presented by**
Dong YUAN

School of Electrical and Computer Engineering

dong.yuan@sydney.edu.au

THE UNIVERSITY OF SYDNEY



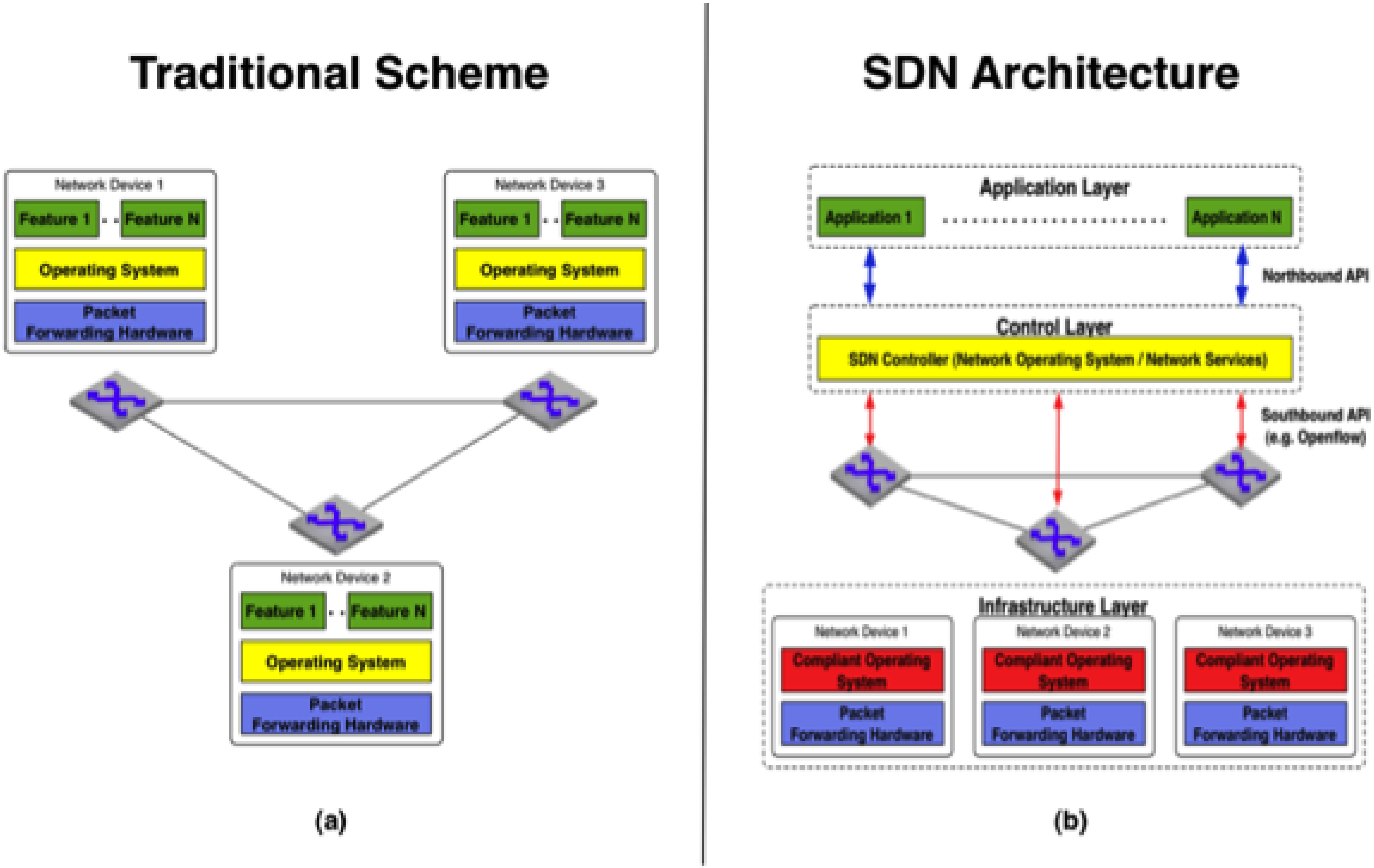
# Contents

– Openflow Switch

– Control and Data Separation
  – Routing and Datacenter network

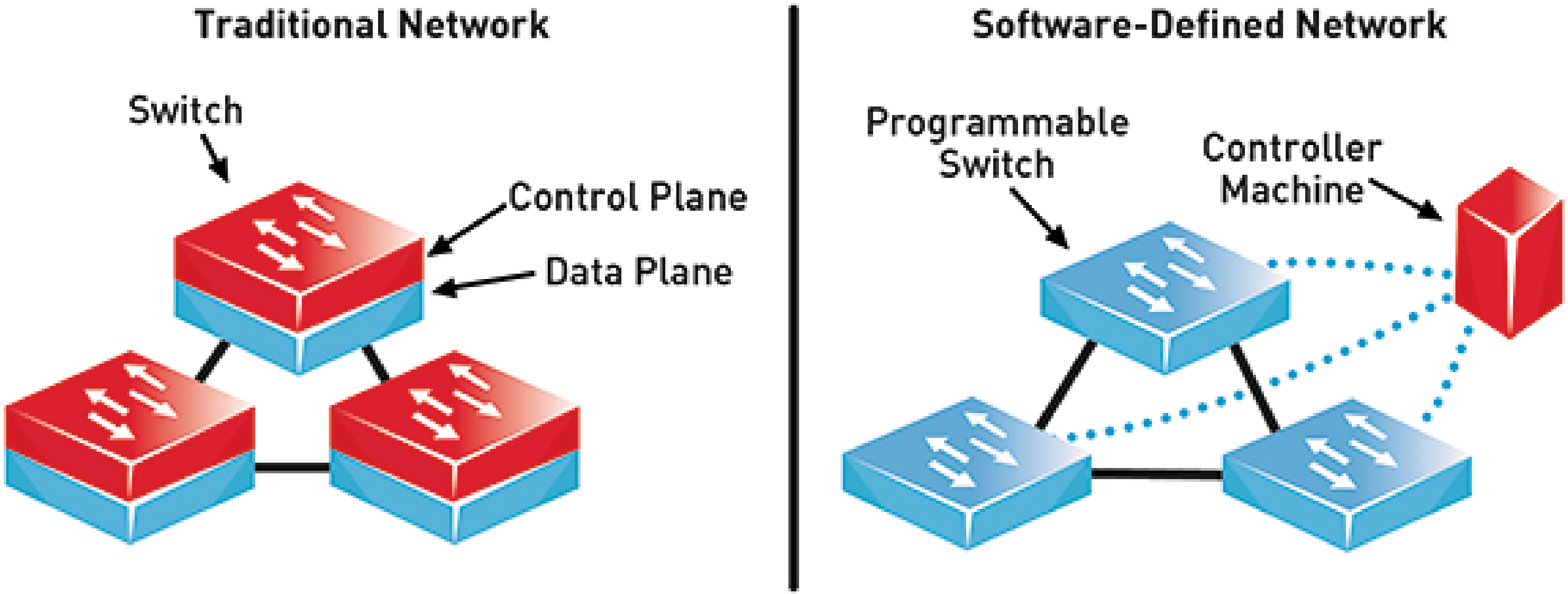
– SDN Challenges

# What is SDN?



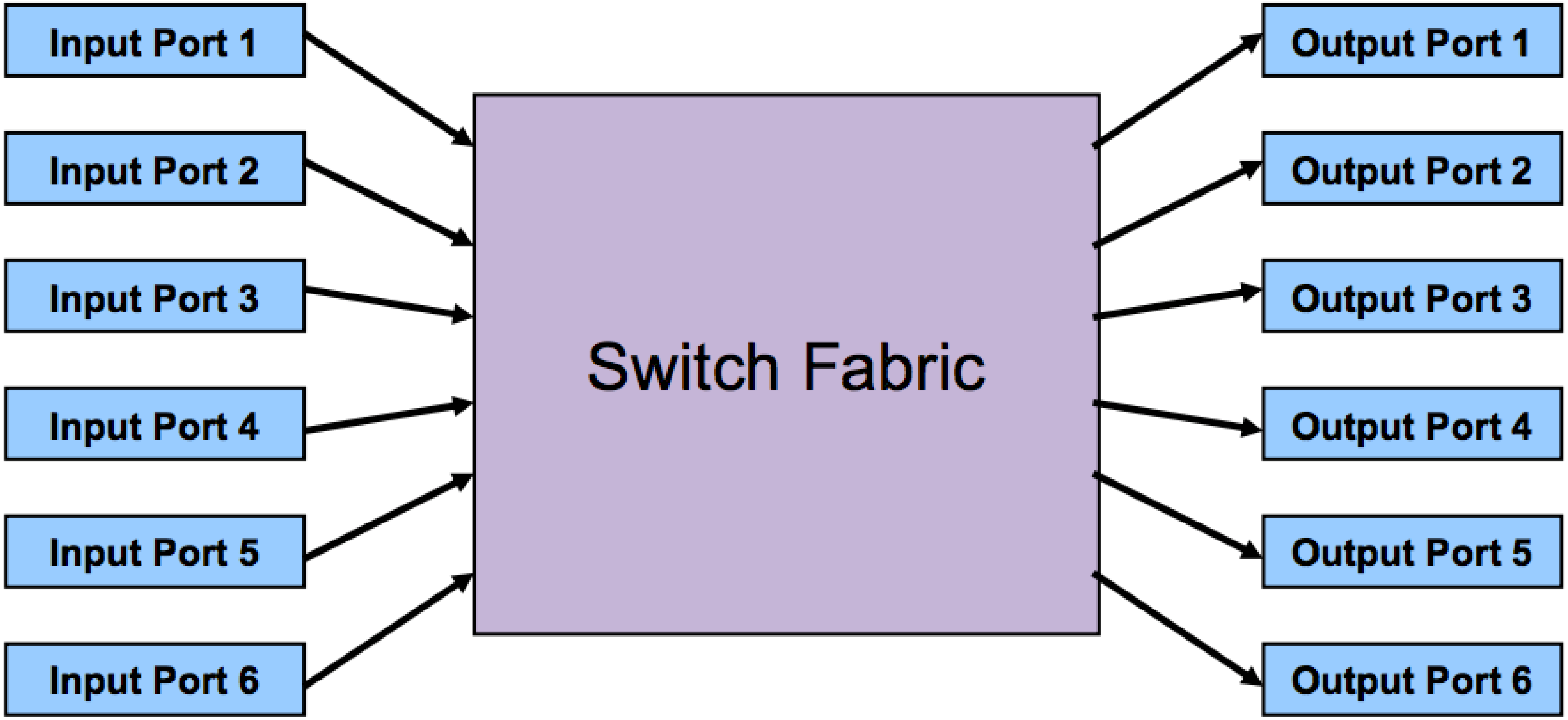

Comparison

# Key difference



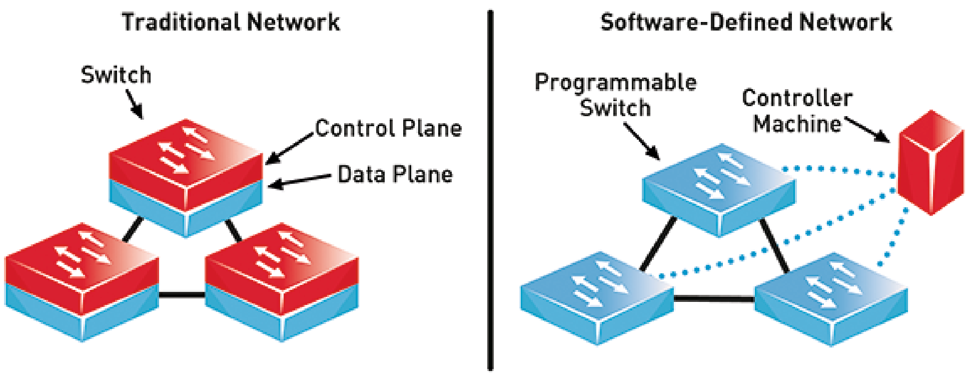

– *Separation of data plane and control plane*

# Switch


Input Port 1
Input Port 2
Input Port 3
Input Port 4
Input Port 5
Input Port 6
Switch Fabric
Output Port 1
Output Port 2
Output Port 3
Output Port 4
Output Port 5
Output Port 6

# SDN Switch

- Flow table - “most important feature of openflow”
  - Match criteria: source, destination, protocol, etc
    - Defined and added dynamically (so called SDN)
  - Actions:
    - flow to which port
    - sending packet to controller
    - Drop a packet,
    - Modify packet header, etc.
  - Priority: for match criteria
  - Counter
  - Time out period:
    - Hard TO, Idle TO




- Connection (TCP) to the Controller
  - Different types of controllers: onos, floodlight, etc.

# How does a flowtable work?

## Example:

| Header Fields | Counters | Actions | Priority |
|---|---|---|---|
| If ingress port == 2 | | Drop packet | 32768 |
| if IP_addr == 129.79.1.1 | | re-write to 10.0.1.1, forward port 3 | 32768 |

Each Flow Table entry has two timers:

**idle_timeout:** seconds of no matching packets after which the flow is removed zero means never timeout
**hard_timeout:** seconds after which the flow is removed zero mean never timeout

If both **idle_timeout** and **hard_timeout** are set, then the flow is removed when the first of the two expires.

# Openflow switches

- Whitebox Switch
  - White-box switches have developed rapidly in the past three decades. The Open Networking Foundation (ONF), the Linux Foundation, the Open Compute Project (OCP), the Telecom Infra Project (Telecom Infra Project, TIP), and other open-source organizations have made significant contributions.
  - White box switches refers to the ability to use ‘generic,’ off-the-shelf switching (or white box switching) and routing hardware, in the forwarding plane of a software-defined network (SDN).
  - They represent the foundational element of the commodity networking ecosystem
  - A common operating system for white box switches is Linux-based because of the many open and free Linux tools available that help administrators customize the devices to their needs.

# Whitebox Switches

- In 2008, Linux began to combine with switching chips to provide large-capacity, high-bandwidth intra-domain data transmission services in data center scenarios.
- In 2011, based on switch software technology, OCP and other organizations began to pay attention to switch virtualization technology. They started the standardization of switch hardware white box.
- In 2015, OCP successfully launched the first white box switch, Wedge.
- Since 2016, technologies such as white-box equipment, software operating systems, and network automation have developed vigorously.

Facebook's 6-pack switch:
https://www.youtube.com/watch?v=OyeoTPSn7b4


https://www.fibermall.com/blog/white-box-switch.htm

# What is data/control separation?

- Control Plane
  - Logic for controlling forwarding behavior
  - E.g., routing protocols, network middlebox configuration (fireware, load balancer).

- Data Plane
  - Forwarding traffic according to data plane logic
  - IP forwarding, Layer 2 switching

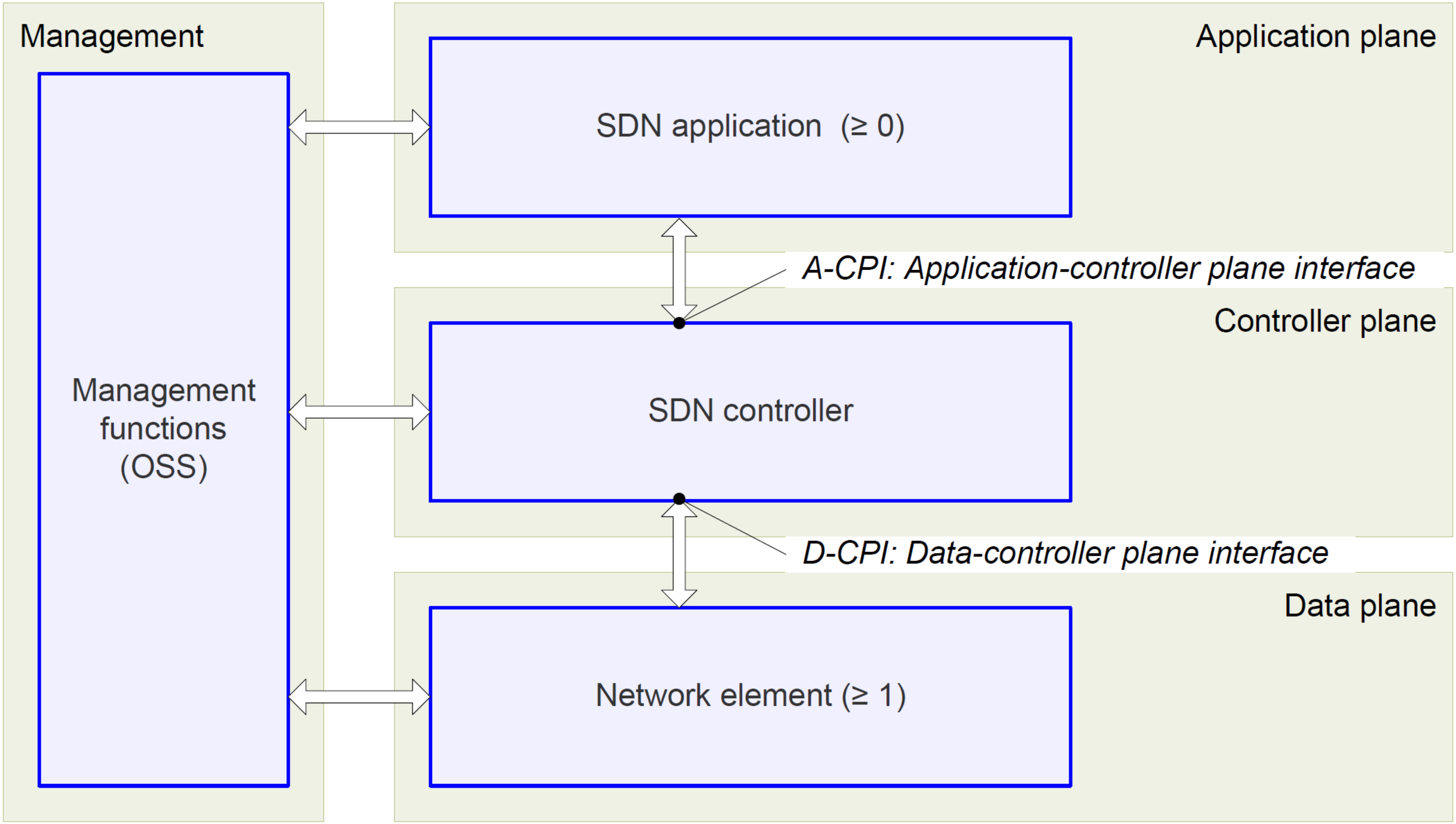
# Principles and architectural components

- Decoupling of controller and data planes
  - Interface between SDN controller and network element is defined in such a way that the SDN controller can delegate significant functionality to the network elements (NE), while remaining aware of NE state.

- Logically centralized control
  - A centralized controller has a broader perspective of the resources under its control and can potentially make better decisions about how to deploy them.

- Exposure of abstract network resources and state to external applications
  - Applications may exist at any level and the distinction between application and control is not precise

# SDN components with management

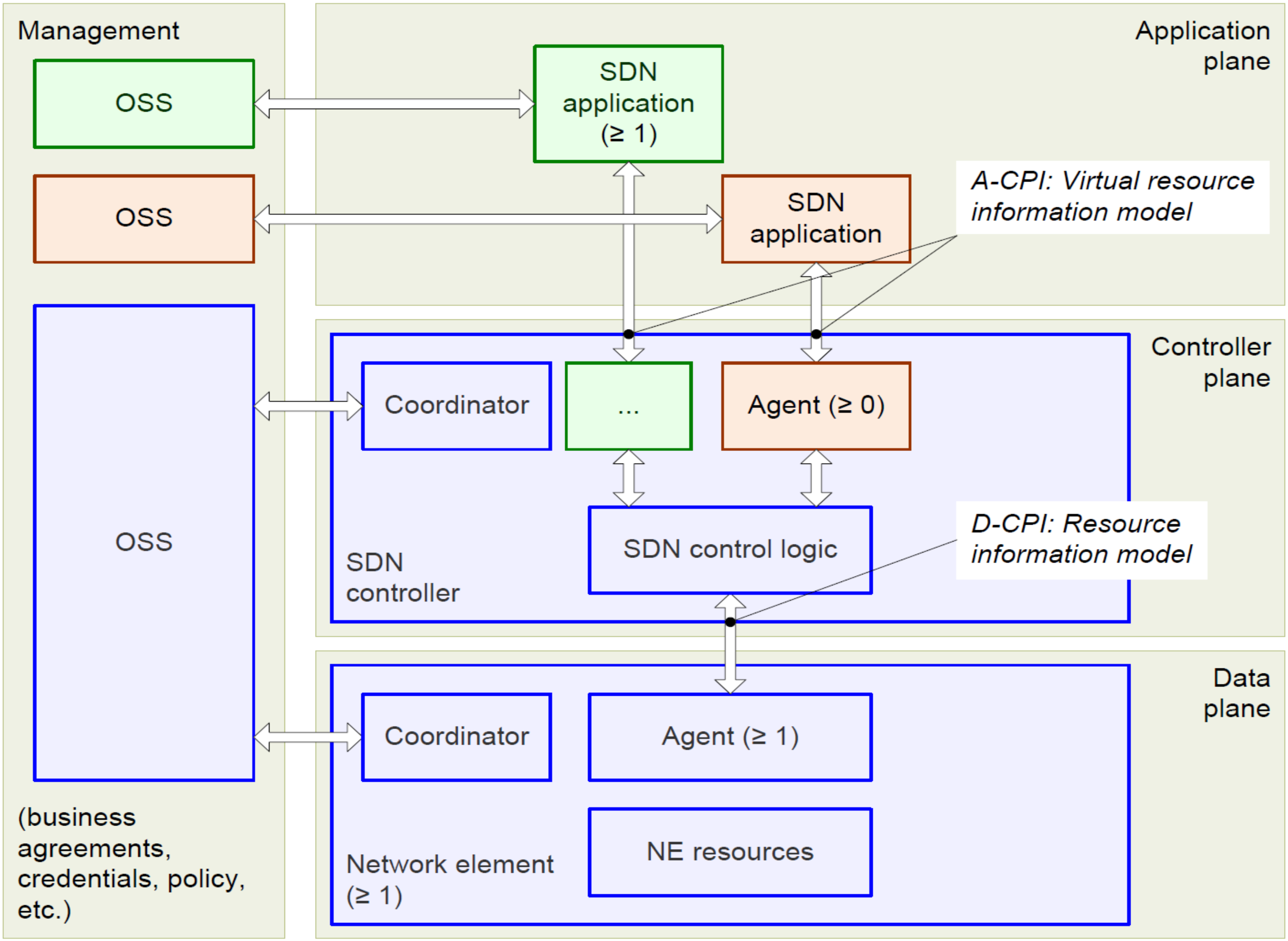


# Reasons for the separation

- Independent evolution and development
  - The software that controls the network can evolve independently of the hardware
  - Easier hardware upgrade

- Control from high-level software program
  - Control behavior using higher level program
  - Debug/check behavior more easily

# SDN architecture extended



# Some usage scenarios

- Routing
  - More control to the decision logic

- Data Centers
  - VM migration, Layer 2 routing

# Inter-domain Routing: wide area

- Inter-domain routing protocol (Constrained Policies)
  - BGP: Border Gateway Protocol, Artificially constrains on routes
  - Route selection is based on a fix set of steps
  - There are limited knobs to control inbound/outbound traffic
  - Very difficulty to incorporate other information (e.g., auxiliary information, time of day).
- Instead: Route controller can directly update state
  - RCP: routing control platform
  - AT&T IRSCP (commercial RCP)

Caesar, M., Caldwell, D., Feamster, N., Rexford, J., Shaikh, A. and van der Merwe, J., 2005, May. Design and implementation of a routing control platform. In *Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation (NSDI)-Volume 2* (pp. 15-28). USENIX Association.

# Routing Examples

- Maintenance of devices
  - Change the default route to bypass the maintained device

- Customer controlled egress selection
  - Multiple way to reach the same destination
  - Giving customer control over the decision

- Better inter-domain routing security
  - Anomaly detection to detect bogus routes
  - Prefer familiar routes over unfamiliar

# Example: Data Centre (Amazon)

- 50,000-80,000 servers, Millions of VMs
  - Any to any, huge inner host links
  - Sub-second migration, guaranteed consistency

- How to keep the synchronization?
  - How to let all switches know about the migration
  - Solution: program switch from a central database

- Cost-effective: Some vendor switches are much more expensive than commodity switch

- More flexible control
  - Tailor network for services
  - Quickly improve and innovate

# Data Center Addressing

- How to address host in a data center
  - Layer 2: less configuration and administration, but bad scaling properties.
    - Ethernet, flat topology. Poor scaling due to broadcast
  - Layer 3: can use the existing routing protocol, but high administration overhead.
    - Intro-domain OSPF(Open Shortest Path First ), operator need to adjust many things

How to get best of both world?

- Separate Controller
  - Topology is dependent on MAC addressing
  - IP addressing for application compatibility

# Other benefits for data centers

– Standard API for network provisioning (i.e. orchestration)

– Integration with VM-based switches (e.g. Open vSwitch)

– New network behaviors that permit scaling to million-VM data centers

– Potential for ODMs to provide more cost effective solutions

# Other Examples:

- Dynamic access control
- Seamless migration for mobility
- Server load balance
- Network virtulisation
- Using multiple wireless access points
- Energy efficient networking
- Adaptive traffic monitoring
- DoS attack detection

http://archive.openflow.org/videos/

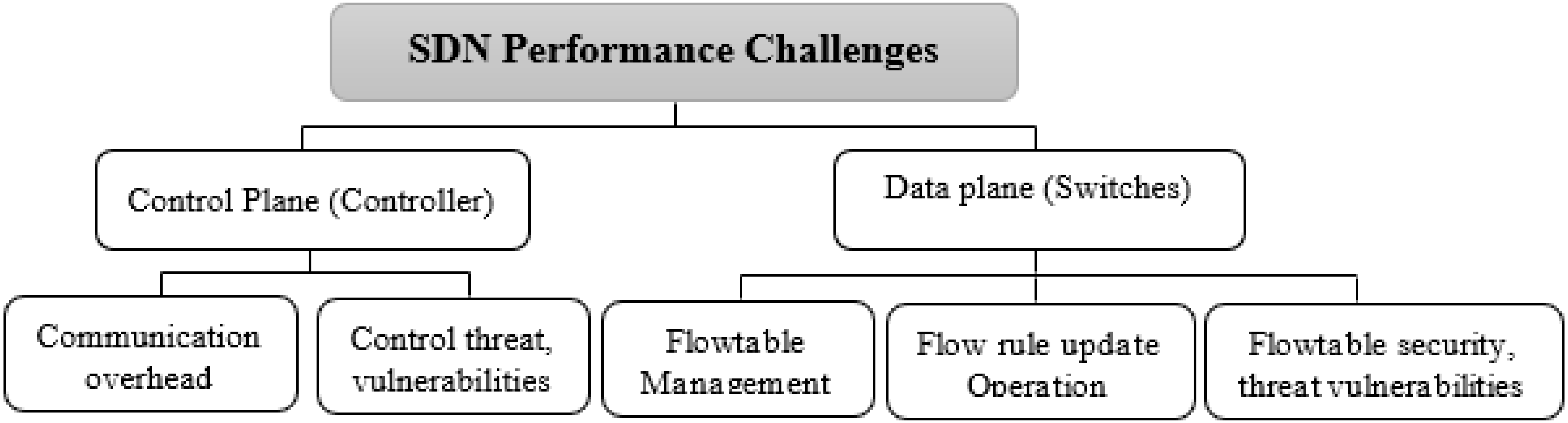
https://www.opennetworking.org/

# Challenges of SDN (potential cons)

- Scalability
  - Control element is responsible for many forwarding elements (often thousands).
  - Problem: must store routes and compute routing for every router
  - Potentially thousands of routers

- Reliability and Security
  - What happens when a controller fails or is compromised?
  - Consistency issue
    - Ensuring consistency across multiple control replicas

# Challenges of SDN




Isyaku, Babangida, et al. "Software defined networking flow table management of openflow switches performance and security challenges: A survey." *Future Internet* 12.9 (2020): 147.

# Scalability

- RCP
  - A Routing Control Platform
    - Centrally manage routing
  - Eliminate redundancy
    - Store a single copy of each route
    - Avoid redundant computation
  - Accelerate lookups
    - Maintain indexes to identify affected routers
  - Performance less types of routing
    - Only perform BGP routing

Caesar, M., Caldwell, D., Feamster, N., Rexford, J., Shaikh, A. and van der Merwe, J., 2005, May. Design and implementation of a routing control platform. In *Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation (NSDI)-Volume 2* (pp. 15-28). USENIX Association.

# Scalability

- ONIX
  - A Distributed Control Platform for Large-scale Production Networks
  - Partitioning: only keep a subset of the overall network information base (NIB) in memory
    - Two different consistency models
      - Strong vs weak
  - Aggregation: Use of hierarchy (e.g., Onix controllers per department or building).
    - Combine statistics, topology information

Koponen T, Casado M, Gude N, Stribling J, Poutievski L, Zhu M, Ramanathan R, Iwata Y, Inoue H, Hama T, Shenker S. Onix: A distributed control platform for large-scale production networks. In OSDI 2010 Oct 4 (Vol. 10, pp. 1-6).

# Reliability

- Replicate RCPs ("Hot Spare")
  - Run multiple identical servers

- Run independent replicas
  - Each replica has its own feed of routes
  - Each replica receives the same inputs and runs the same routing algorithm
  - No need of a consistency protocol in this case

If different replicas see exactly the same information.

# Potential Consistency Problem

If the replicas see different information:

- "Bouncing route"

Solution:

- Route assignment must be consistent
  - Even in presence of failures and partitions
- Fortunately
  - Flooding-based IGP (e.g., OSPF) means each replica knows which partitions it connects to

# Why consistent

- Single RCP under partition
  - Only use state from routers partition in assigning its routes
  - Ensures next hop is reachable

- Multiple RCPs under partition
  - RCPs receive same state from each partition they can reach
  - IGP provides complete visibility and connectivity
  - Only acts on partition if it has complete state

# Thank you!

References:

https://www.scs.gatech.edu/news/195201/free-online-sdn-course

https://courses.engr.illinois.edu/cs438/sp2010/slides/lec07_switches.pdf

https://www.usenix.org/conference/osdi10/onix-distributed-control-platform-large-scale-production-networks

https://www.sdxcentral.com/cloud/converged-datacenter/whitebox/definitions/what-is-white-box-networking/

THE UNIVERSITY OF SYDNEY