Software Defined Networks 101

Bryan Ng and Ian Welch,
Victoria University of Wellington
10 February 2016

TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

Talk

- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

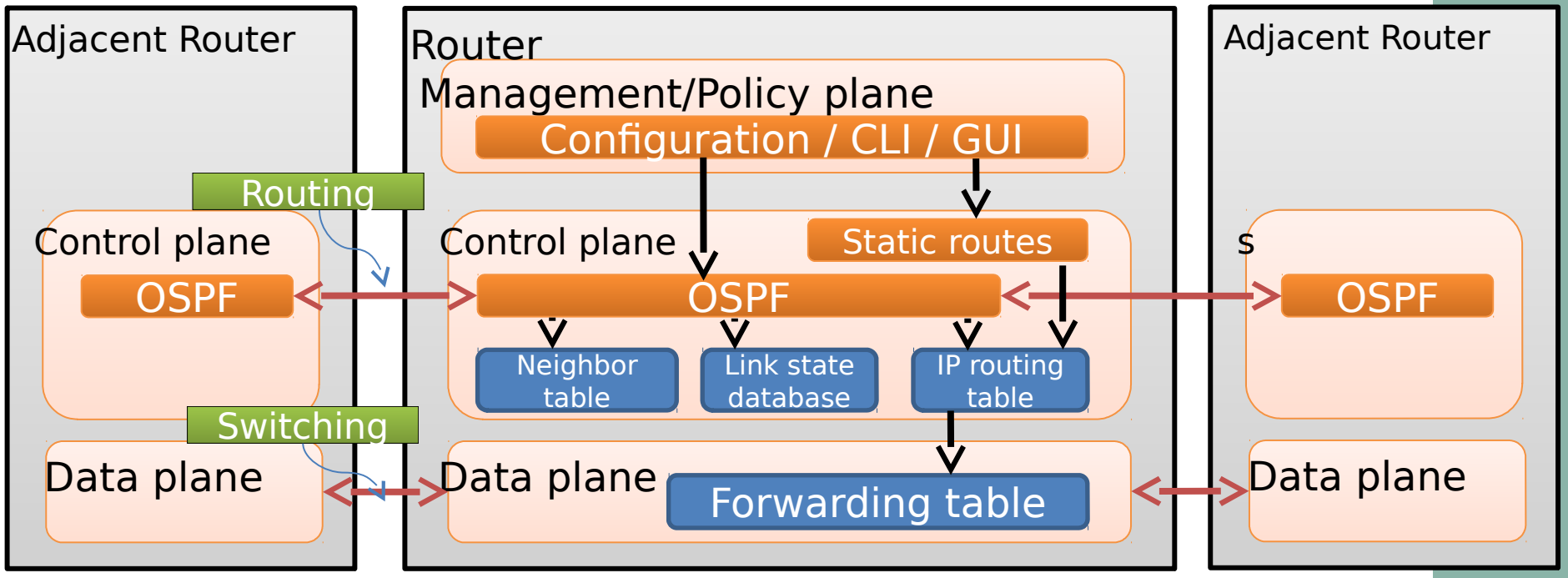
Traditional network architecture

- Heriarchial tiers of ethernet switches connect hosts to form local area networks (layer 2)
- Local area networks connected by routers (layer 3)
- Organisations use gateways (layer 3) to connect to their Internet providers.

Assumes static network perimeters, limited client and server mobility.

Traditional network node: Router

Changing protocols is hard, monolithic implementation, vendor specific protocols



Shamelessly copied from ONF© , J Rexford and Chao HC with permissions.

Changing assumptions

- *Data centre:*
  - servers appear and disappear (essentially are mobile)
- *Bring your own devices:*
  - traditional network control is coarse grained, need fine-grained control
  - cannot rely on fixed perimeters of control
- *Cloud computing and big data:*
  - we can scale compute, want to be able to scale bandwidth as well

Subsequent problems

- Static nature of networks make it difficult to adapt to changing mobility.
- Diversity of network devices and ways to configure them make it hard to enforce system wide policies.
- Can no longer rely on overprovision to cope with increased dynamic bandwidth requirements.
- Monolithic architectures cannot innovate faster than the three year standard product cycle.

Talk

- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

Software Defined Networking

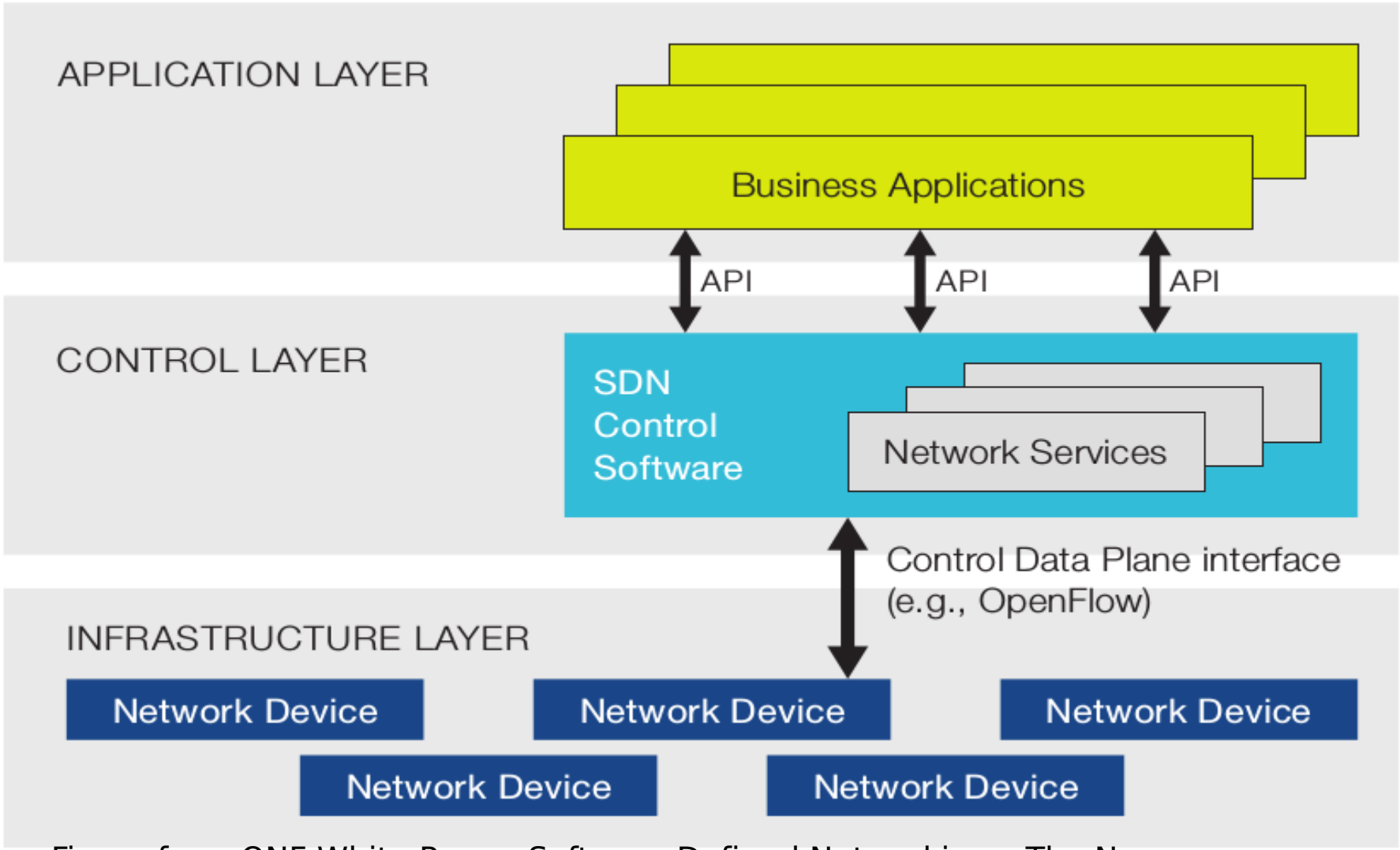


Figure from ONF White Paper: Software Defined Networking – The New Norm for Networks (2011)

Software Defined Networking

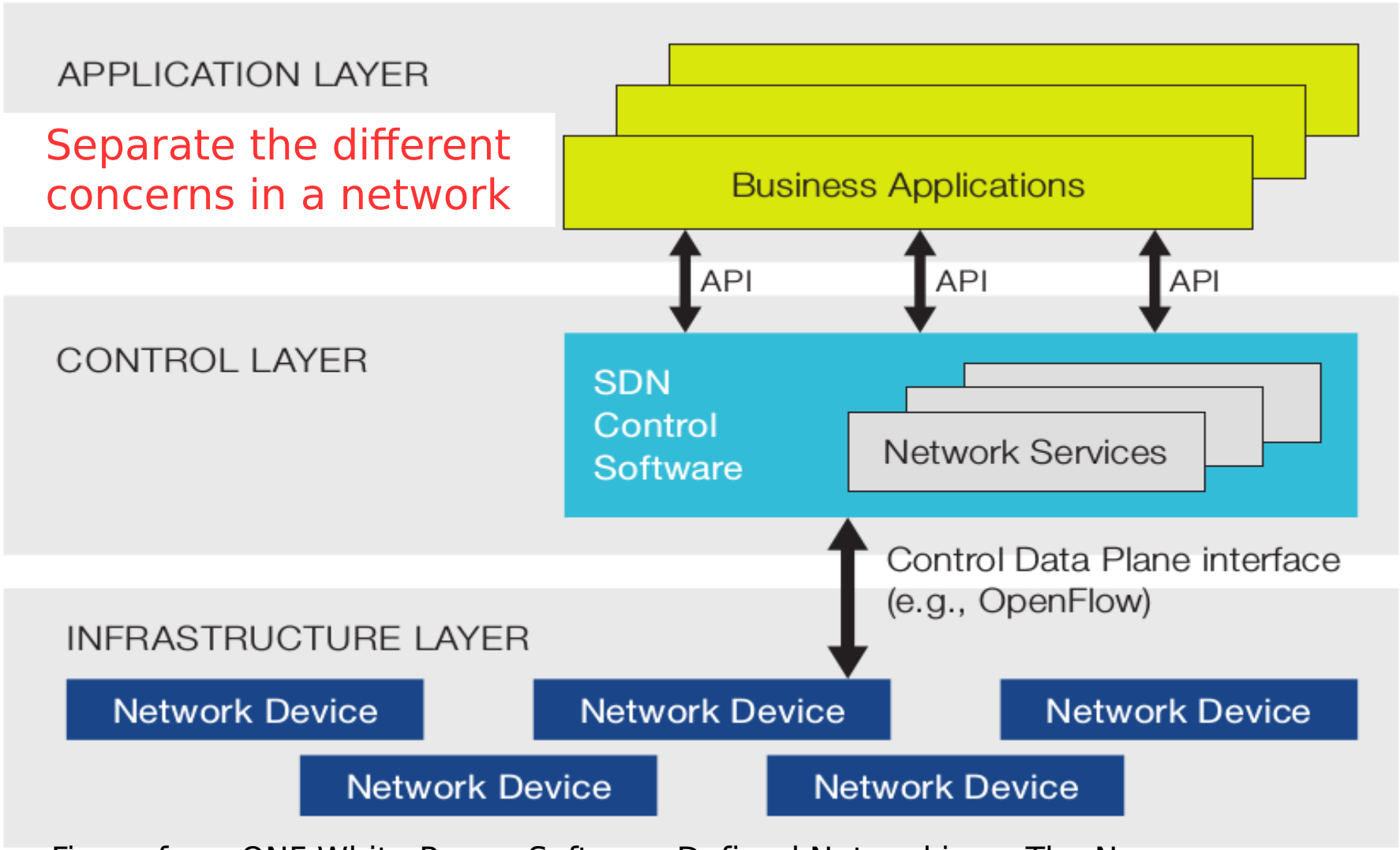


Figure from ONF White Paper: Software Defined Networking – The New Norm for Networks (2011)

Software Defined Networking

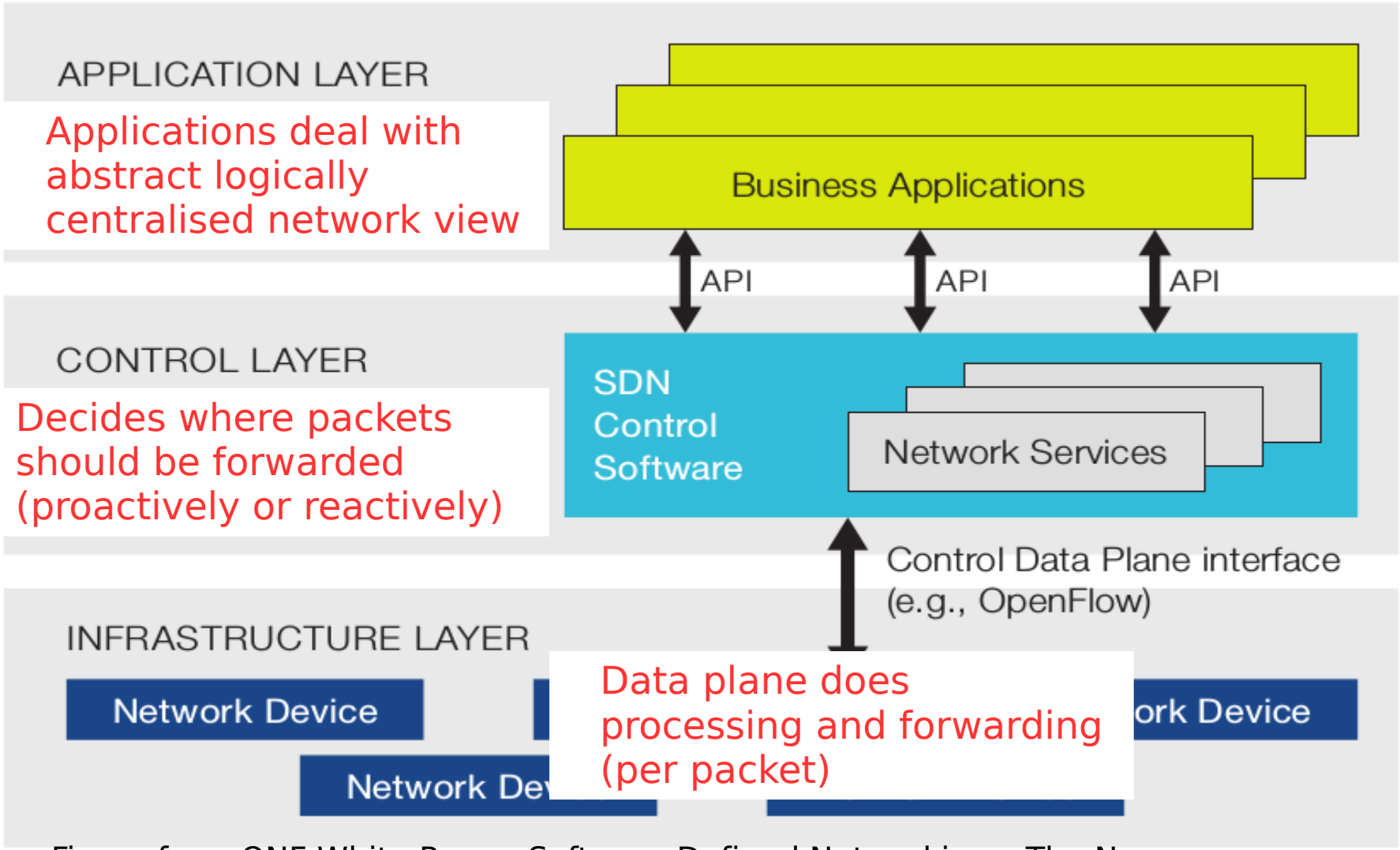


Figure from ONF White Paper: Software Defined Networking – The New Norm for Networks (2011)

Potential benefits

- Directly programmable network (decoupling allows control logic to be changed independently of forwarding hardware).
- Agile network (can reprogramme network flow to meet needs).
- Centrally managed (allowing optimal choices).
- Programmatically configured (allow system-wide enforcement of policies).
- Open standards-based and vendor-neutral (avoiding vendor tie-in)

SDN varieties

- Openflow SDN
- Vendor X SDN
- Network functions virtualisation
- NetFPGA-based SDN
- Hybrid SDN solutions

Talk

- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

Origins of OpenFlow

- Desire for clean slate experimentation.
- Manufacturers opening up forwarding behaviours.
- Broader number of manufacturers wanting to innovate.

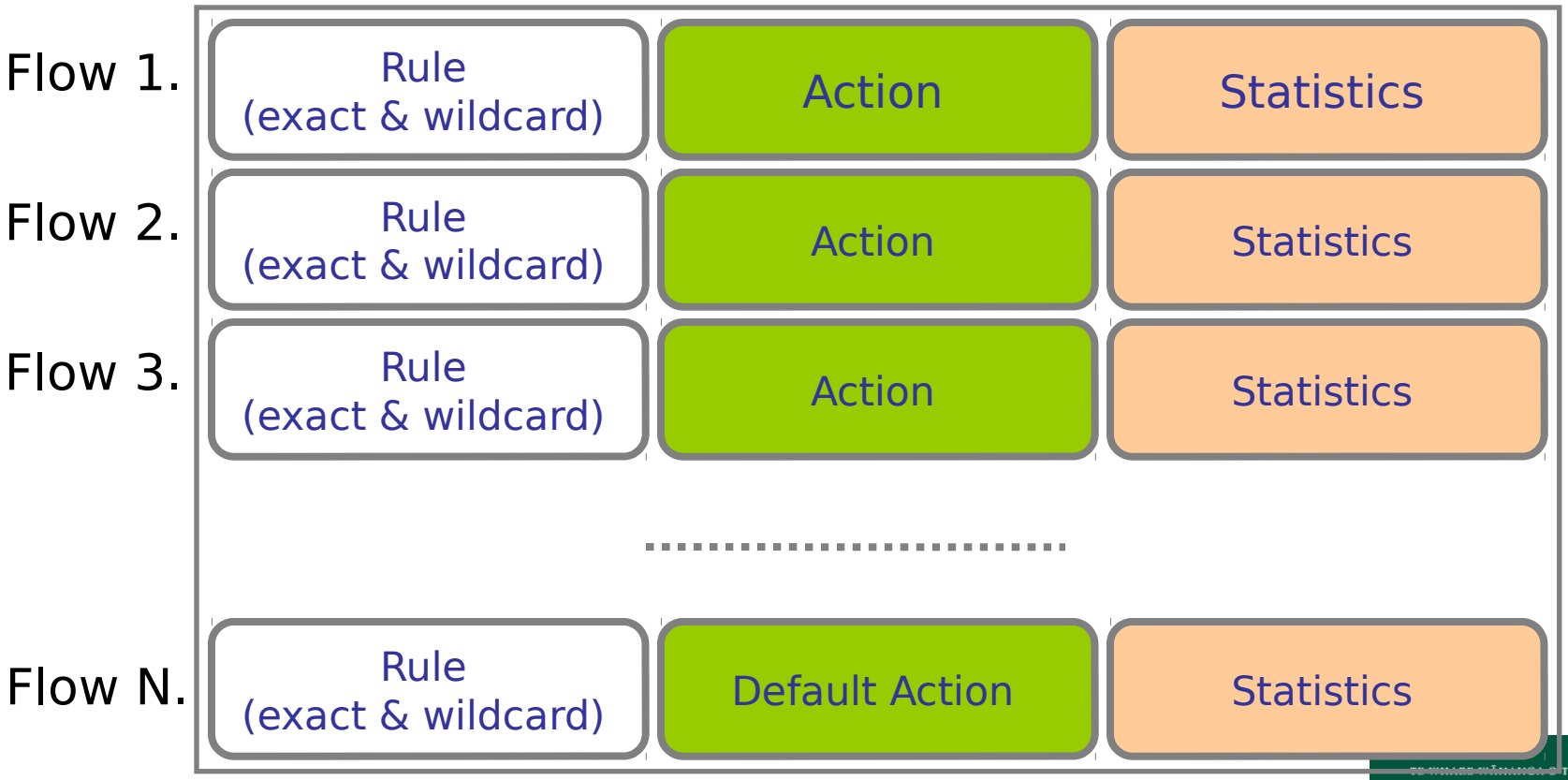
- Openflow standardised API (Open Networking Foundation) for controlling forwarding behaviour of switches.
- Unifies a broad range of network devices (routers, learning switches, NATs …).

OpenFlow Versions

- Openflow 1.0 allowed initial experimentation.
- Subsequent versions (1.1, 1.2, 1.3, 1.4 and 1.5) have added extra capabilities:
  - IPv6
  - Multiple tables (goto)
  - Managing full tables
- Following examples are version 1.0

OpenFlow Switch

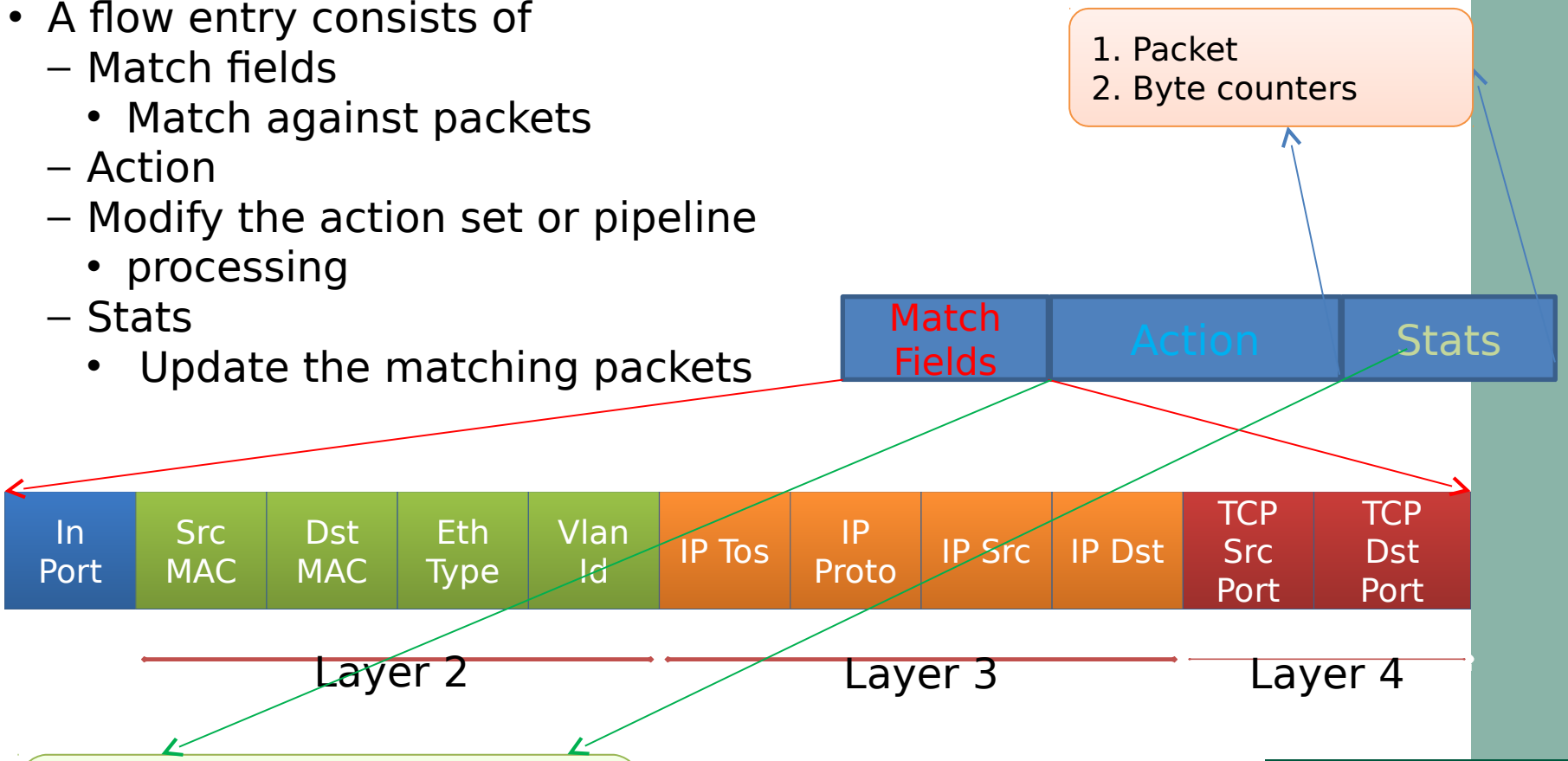
- Performs packet lookup and forwarding



Figures from Chao HC & Y. Liang with permissions.

Flow Entry (OF 1.x)

- A flow entry consists of
  - Match fields
    - Match against packets
  - Action
  - Modify the action set or pipeline
    - processing
  - Stats
    - Update the matching packets



Figures from Chao HC & Y. Liang with permissions.

Example Table

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f:..	*	*	*	*	*	*	*	port6
port3	00:20..	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6
*	*	*	*	*	*	*	*	*	22	drop

More exact matches have priority.

Table miss sends packet to controller over secure TLS connection.

Table entries have idle and hard timeouts.

O-65,535 seconds (0 = no timeouts).

Switch → Controller

- PACKET_IN:
  - Rule not found in table.
  - Encapsulated packet (metadata).
- FLOW_REMOVED:
  - Due to timeout or controller driven removal.
- PORT_STATUS:
  - Port events.

Controller → Switch

- Modify switch state:
  - Add, remove and modify entries.
- Read statistics:
  - Flow tables, ports and individual flow entries.
- Barriers:
  - Synchronisation primitives.

Controller → Switch

- Modify switch state:
  - Add, remove and modify entries.
- Read statistics:
  - Flow tables, ports and individual flow entries.
- Barriers:
  - Synchronisation primitives.

Talk

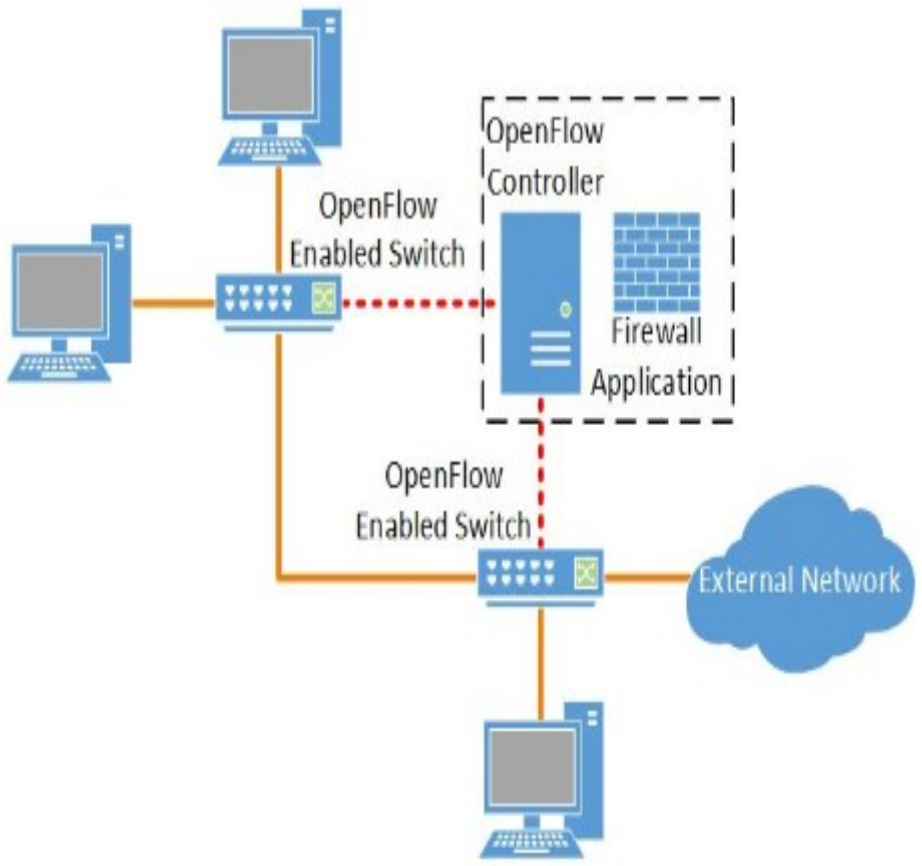
- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

Example: Firewalls

- Context here is a campus or small enterprise network.
- Examples in SDN session about traffic shaping.
- Here focus on implementing a firewall (why? honours project completed last year, illustrates some tradeoffs and was a learning exercise for us).

Example: Access Control

- Implement time-based access ACLs within a network.
- Firewall application.
- Administrator uses simple policy language to specify rules.
- Enforced globally across organisation.



Example: Access Control

- Policy translation to flow table entry.
  - only allow specific source and destination endpoints
  - drop all other flows
- Proactive distribution of rules (avoid the slow path).
- Housekeeping when new switches appear and disappear.

User specified.

Stored on the controller during runtime.

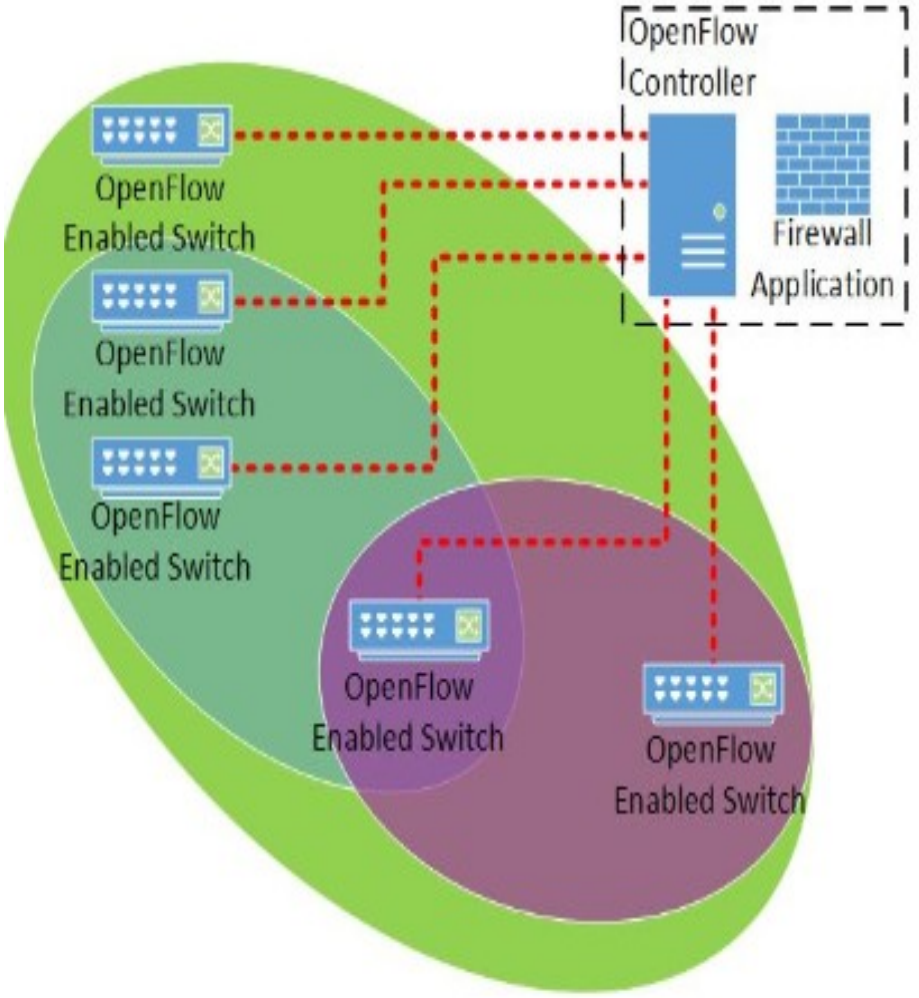
Stored on an OpenFlow switch.

Example: Access Control

- Add in time into the equation (consider schools).
- Timeout range not sufficient to express desired constraints.
- Controller proactively removes flow table entries.
- *Question: what would effect of propagation delay be in a larger organisation? Inconsistency across switches?*

Example: Access Control

- We decided to enforce rules in the core.
- Does that make sense? Why waste table space?
- Defined a policy language to group switches in shared policy domains.



Example: Access Control

- Implemented in software and tested using mininet.
- Mininet is a network emulator allowing topologies to be created made up of openflow switches, controllers and hosts.
- Didn't test using a real switch (other students have done that) and this would be part of any extension.

Example: Access Control

- Use SDN to manipulate flow rules according to policy definitions.
- Pseudocode ~
  - User connect and attempt to send pkts
  - No policy … …. Send to controller
  - Controller looksup policy DB … he’s our CTO … super user
  - Install flow rules for CTO …
  - May want to log his info?

Talk

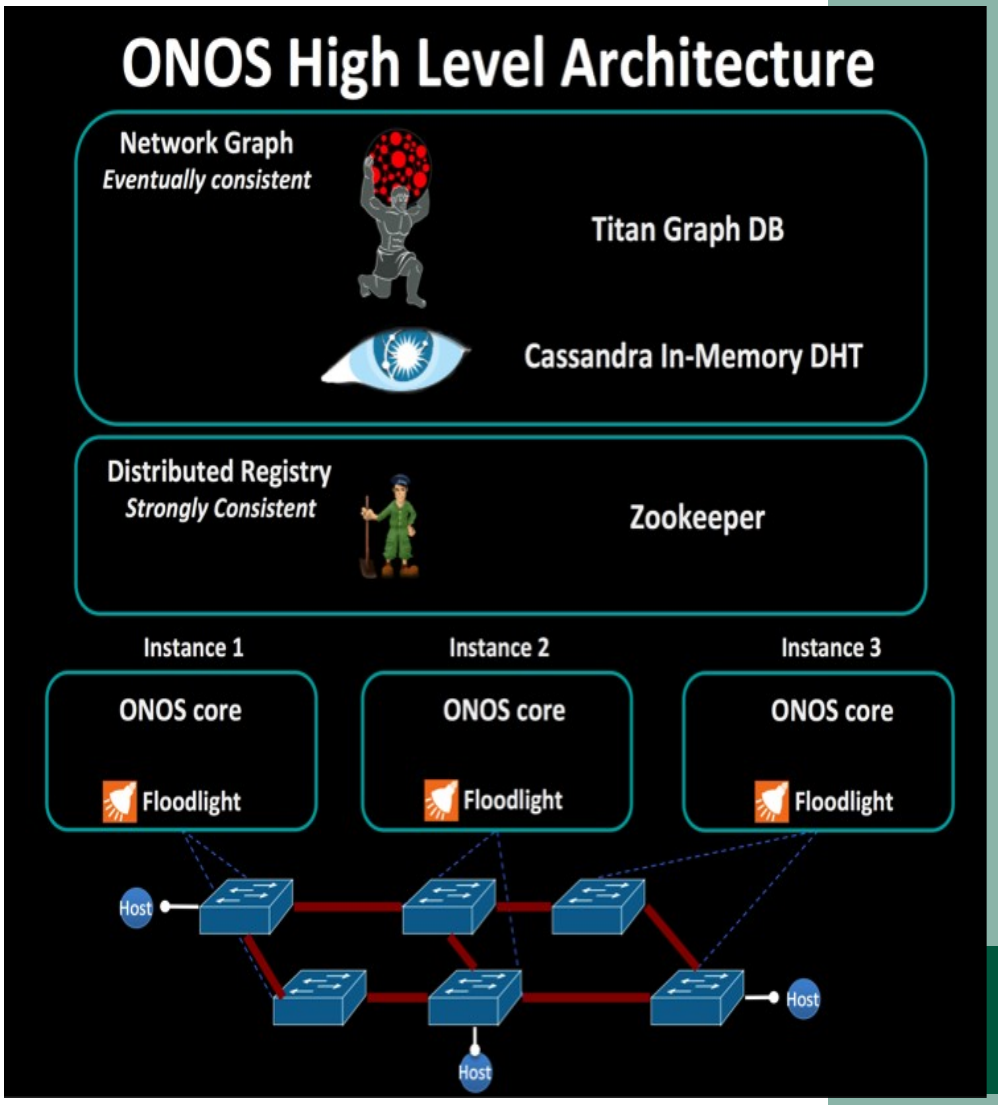
- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

OpenFlow Switches

- Openflow switches are widely deployed in big organisations and are reliable (Google, Microsoft, ESnet etc.)
- We use mininet/ovs for classes (no need for hardware) but that is different from experience of real equipment.
- We use cheaper or borrowed equipment and there is a lot evolution is going on.
- Can mean that things don't work as expect or there are limits of performance.
- More generally, some parts of specifications are optional so you need to be aware of this!

Software Controllers

- Differ in terms of language, version
  of open flow supported
  and features.
- Examples:
  - Ryu (Ree-yooh) is component based SDN programming framework written in python.
  - Floodlight is SDN controller written in Java.
  - ONOS aims to provide a reliable carrier-grade controller that implements a network operating system.



Talk

- Problems with traditional networks
- What is SDN and how it helps
- Openflow SDN
- Example: Firewalls
- Switches and Controllers
- Hands on Session

Hands on Session

- We've adapted some existing tutorials covering:
  - Using mininet and ovs tools
  - Running a ryu sdn application
  - Using the faucet switch
  - Using two sdn applications together

- Self-paced, we will get you up and running and circulate.