Software Defined Networking (SDN)

COSC349—Cloud Computing Architecture

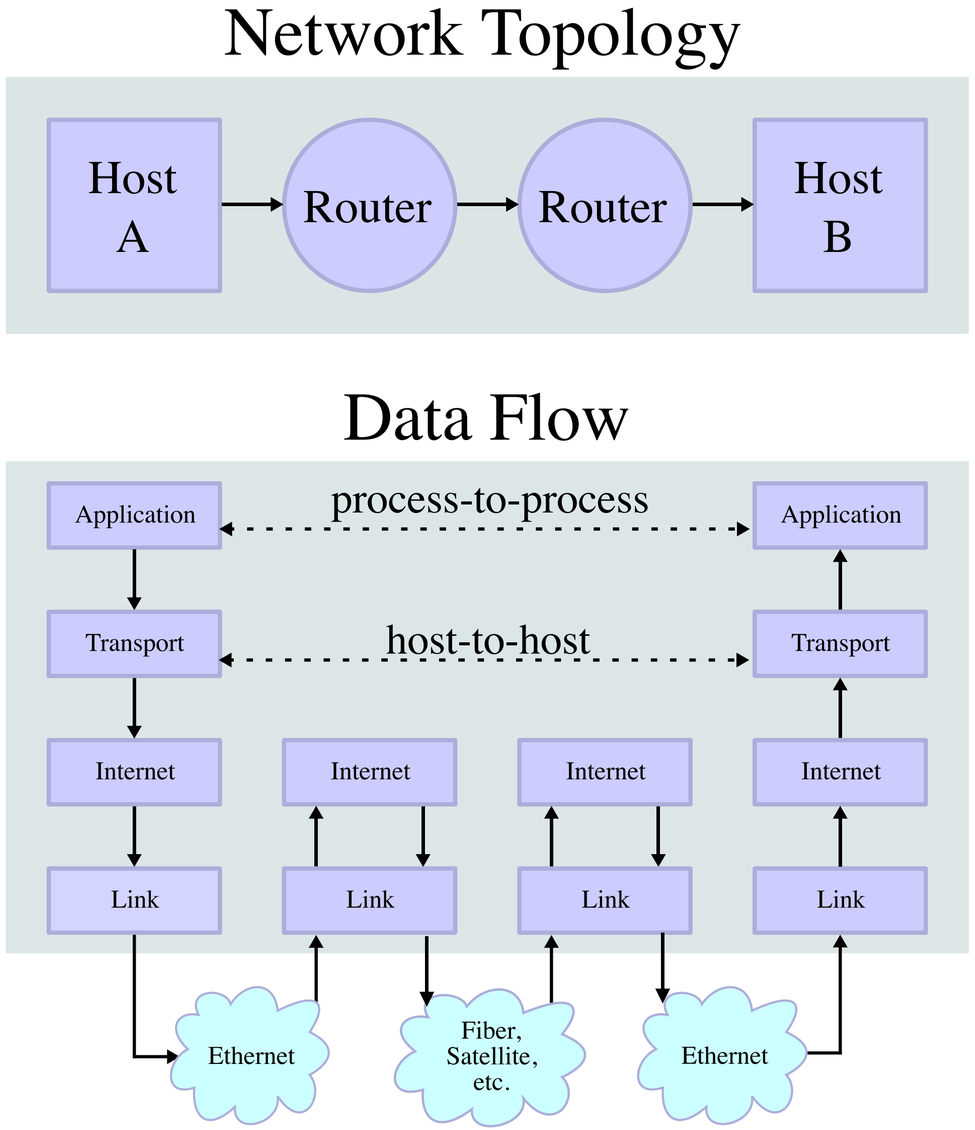
David Eyers

Learning objectives

- Outline goals of **software defined networking** (SDN)
- Describe why SDN is of **interest to cloud providers**
- Give examples of services commonly redeployed to use **network function virtualisation** (NFV)
- Explain the trend in (cloud) data-centres toward **programmable network devices** with open designs

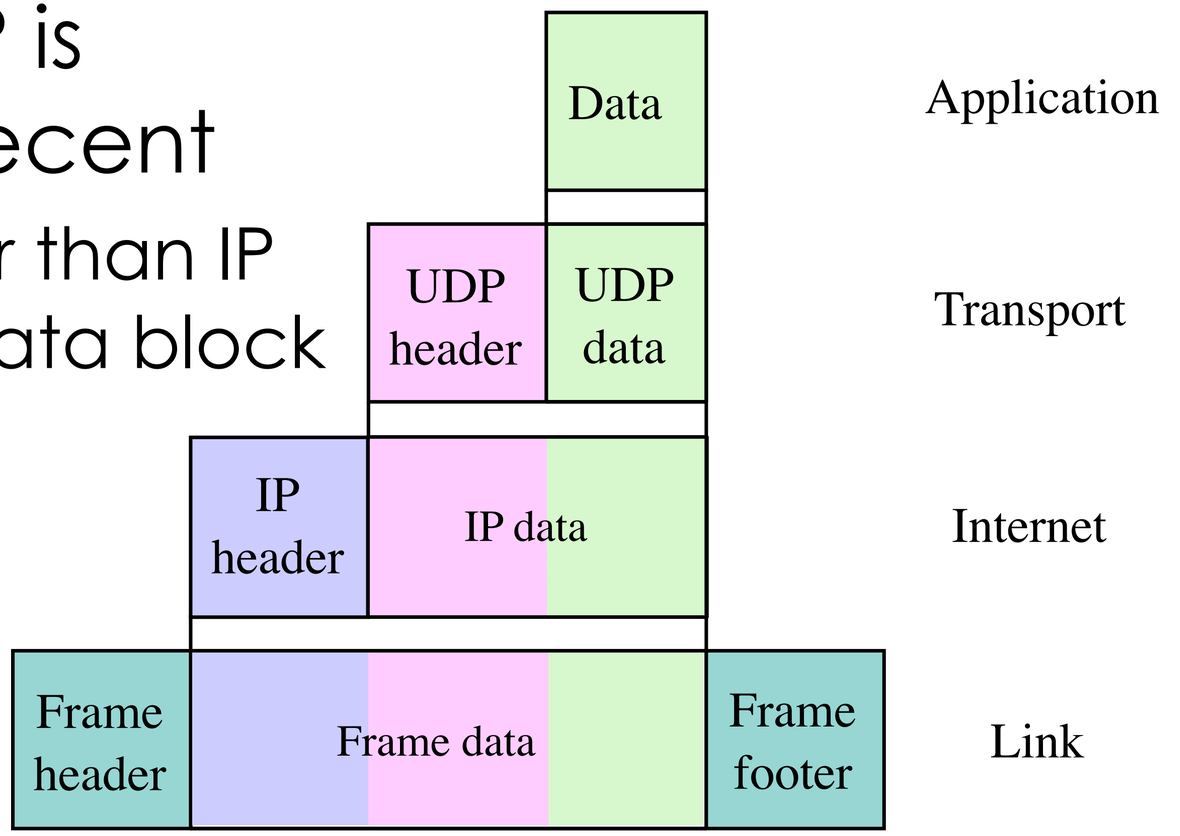
Quick refresher on IP networking layers

- IP networking's success depends in part on **layering**
  - Different layers can change technologies independently
    - *e.g.*, **Wi-Fi versus cable at link level**
- Different devices handle layers
  - **Switches** usually work at link level
  - **Routers** work at IP / transport level
- Layering is not strictly enforced
  - e.g., cross-layer optimisation, VPNs…



Data carried at different network layers

- Data for applications using IP is augmented at each layer decent
  - Ethernet frames' total size larger than IP packets' total size for a given data block
- UDP shown here **transports datagrams** (chunks of data)
- TCP instead **transports coherent streams** of data
  - Includes **retransmissions** and **congestion control**



Ethernet network switch hardware

- Hosts wired to ports of switch
- **MAC addresses in Ethernet frames** examined
  - Ternary content addressable memory (TCAM) looks up MAC:
  - determines which switch port(s) to send Ethernet frame to
- Switch backplane has **higher bandwidth than ports**
  - Needs to allow pairs of ports to communicate in parallel
  - Uplink ports often higher speed than normal ports
- **Switches run software**—need firmware upgrades, etc.
  - Also, virtual switches can be run by VMM—*e.g.*, VirtualBox's

New capabilities for cloud data-centres

- Cloud DCs don't need support for *ad hoc* networking
  - Topology and machines on network are **known and managed**
  - Instead switches ideally integrate **DC-specific control software**

- *E.g.*, ARP—address resolution protocol—**is unneeded**:
  - ARP broadcasts on Ethernet "Which MAC has IP X?" & replies
  - Broadcasts waste DC net. bandwidth, when **answers are known**

- Virtualisation can cause physical server to have many MACs… but deployment software **already knows this set**

Software Defined Networking (SDN)

- SDN dissociates switch's data plane from control plane
  - **Data plane**: high-speed hardware for forwarding data
  - **Control plane**: manages data plane's forwarding paths
- Thence SDN facilitates **custom control plane software**
  - This can be thought of as a form of virtualisation of switch
- SDN typically provides agile, **centralised management**
  - Switches operating independently can be fiddly to manage
- SDN has embodied a push for open standards use
  - … also, **vendor-neutral solutions** within networking hardware

OpenFlow—a popular SDN design

- OpenFlow allows remote management of switch rules
  - OF switches use dedicated network link to a **controller**
    - Controller is often a 'normal' server, *e.g.*, running Linux
    - Important to secure this link—should use TLS
  - Typically for first-time packet forwarding, **call out to controller**
  - Controller provides resulting **packet matching rules & actions**
  - Establishes **flow** to potentially be used for subsequent packets
- OF can be easily **implemented within existing switches**
  - OF controller can co-exist well with existing control software

FYI Faucet: NZ-developed SDN controller

- See faucet.nz: open source project developed in NZ
  - … but while Josh Bailey is in Wellington, he's a core Googler
  - Many Faucet events in Wellington (REANNZ HQ is in WLG too)
- Adds many features beyond basic OpenFlow, *e.g.*:
  - Use of multiple controllers to support **high availability**
  - **Online controller update** and upgrade
  - Integrated **real-time dashboards** and **time-series DB** for logging
  - Policy-based forwarding for offload processing *e.g.*, NFV
  - Port mirroring—*i.e.*, duplication of data down multiple ports

Network Function Virtualisation (NFV)

- Remove network control functions from switch firmware
  - Instead **virtualise software handling NFs** (*e.g.* control protocols)
- Common facilities supported by NFV include:
  - **DHCP**—dynamic host configuration protocol (give out IPs)
  - **Firewalls**—filter and modify traffic to secure networks
  - **DPI**—deep packet inspection: scans packet data
  - **IDS**—intrusion detection systems scan network for attacks
  - **NTP**—network time protocol
- NFV requires careful management and monitoring

FYI: P4—low-level network programming

- Programming Protocol-independent Packet Processors
  - Can **program network equipment** targeting (v)CPUs but also:
    - FPGAs—programmable hardware; network processors and ASICs
  - Much **more abstract than IP**, but of course supports IP
    - also can be applied to Ethernet, MPLS, TCP, etc.
  - Supports **dynamic reconfiguration** of network devices
  - Stateful processing using registers, counters and meters
- P4 further **disaggregates network functions** than SDN
  - P4's founders include SDN founders

Application-level routing implementations

- Consider Internet innovations running at app-level, *e.g.*
  - **BitTorrent**—global-scale, efficient distribution of large objects
  - **HTTP Adaptive Streaming**—video streaming (YouTube, Netflix)
  - (old) Skype—used **peer-to-peer (P2P) routing** to connect VoIP
- A P2P generalisation: distributed hash tables (DHTs)
  - Nodes given numerical IDs, $\mathcal{O}(\log n)$ complexity to reach any ID
  - Builds overlay network over existing IP network
  - Easily support *ad hoc* client connectivity: clients come and go
  - Replication can be done by delivering to neighbourhood of ID

Application-level routing can move to SDN

- SDN helps innovative **application-driven protocols**
  - Application-level routing is slow: traverses multiple net. stacks
  - 'Push down' application software into **SDN implementation**
  - Likewise optimise virtual network switches' stacks
  - Provide **programmability of software** with **speed of hardware**
- Increasing trend toward **name-based networking**
  - Route traffic based on 'topic' or 'content', not IP address
  - Facilitates building spanning trees to disseminate content
  - Can effectively support n–m delivery of network data

SDN in practice

- Google back in 2012 announced its **use of SDN**
  - Reworked company's internal network to use OpenFlow
  - Also use some similar type of management on WAN links
    - Google don't need *ad hoc* configuration support: **has known links**
    - **Google has specific playbook** for what to do when links fail
- P4: industrial collaborators include many large players:
  - Alibaba; Baidu; Cisco; Google; Intel; Microsoft; Tencent; …
  - Provides a strong basis for academic research
    - *e.g.*, use within open network device platforms such as NetFPGA