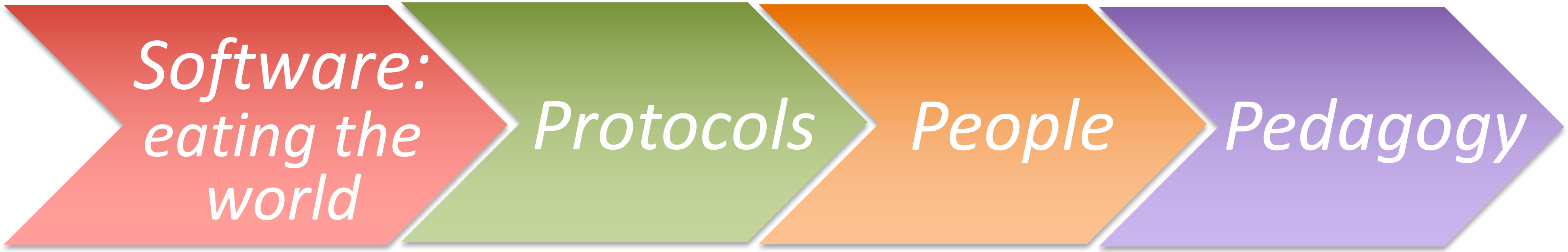
The Software-ization of Networking: Protocols, People, Pedagogy

Jim Kurose

Distinguished University Professor
Associate Chancellor for Partnerships and Innovation
College of Information and Computer Sciences
University of Massachusetts, Amherst

ICETC 2022
November, 2022

Outline



Software: eating the world

Protocols

People

Pedagogy

“Software is eating the world.”

Marc Andreessen
MOSAIC inventor, Netscape co-founder, VC

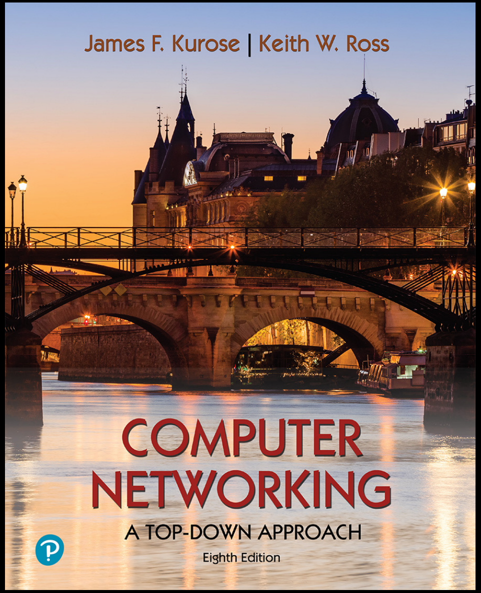
In 2030. .. “Networking students will learn how to program a network top down as a distributed computing platform. Protocols will be described in quaint historical terms.”

Nick McKeown
SDN architect, Stanford, Intel

“Protocols are running *everywhere* in the Internet, and consequently much of this book is about computer network protocols.” (*Computer Networking: A top-Down Approach*, 1st – 8th editions, 1999-2020)

Jim Kurose

Co-author, *Computer Networking: A Top-Down Approach*



Outline

Software: eating the world

Protocols

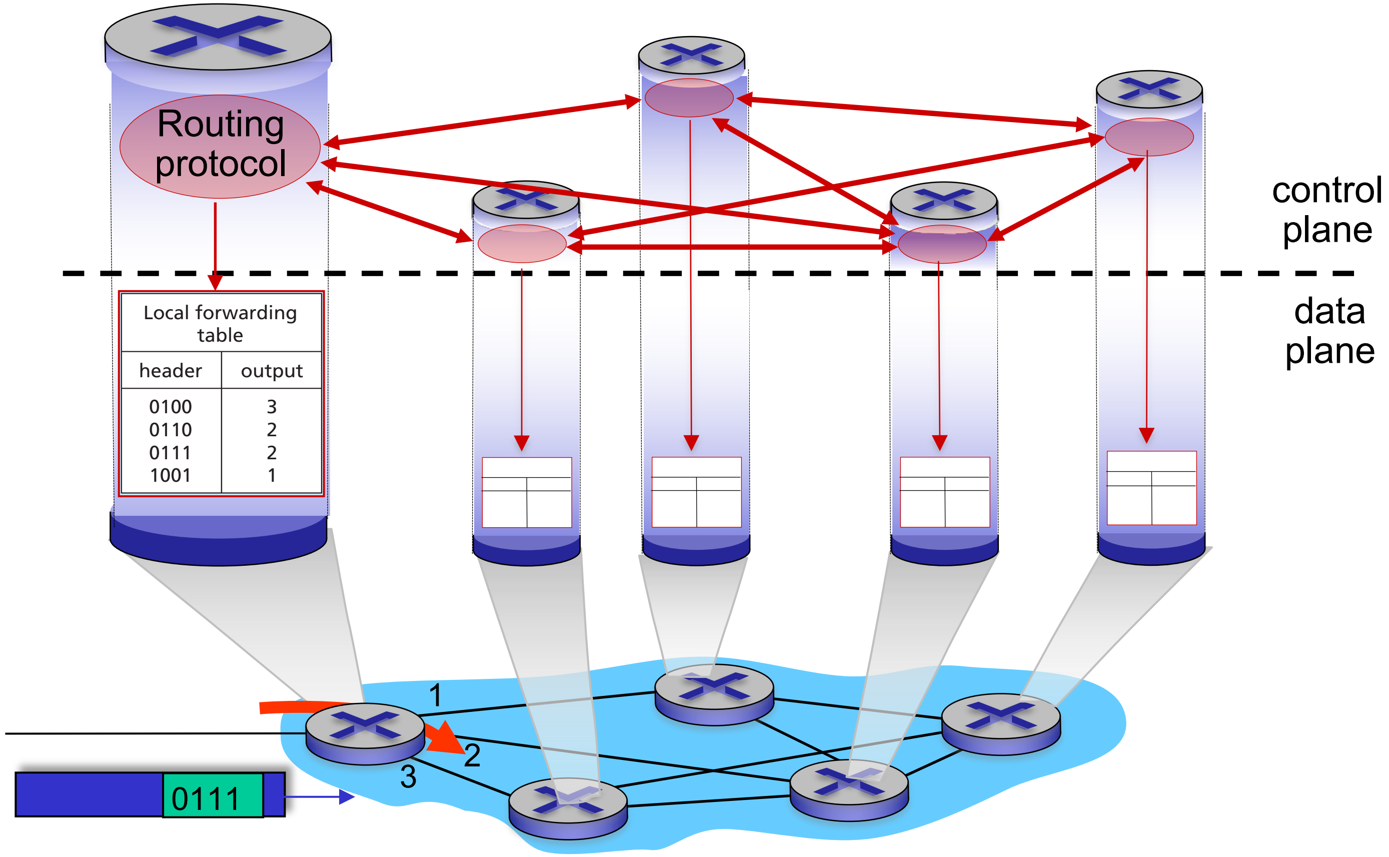
People

Pedagogy

Are protocols dead?

Traditional, in-network control, management

- traditionally: “in network” distributed, local view of network control and management
- many protocols:
  - IS-IS, OSPF, iBGP (Example to right)
  - HTTP, DASH
  - TCP
  - 802.3, 802.11
  - DHCP, ARP, RADIUS SNMP

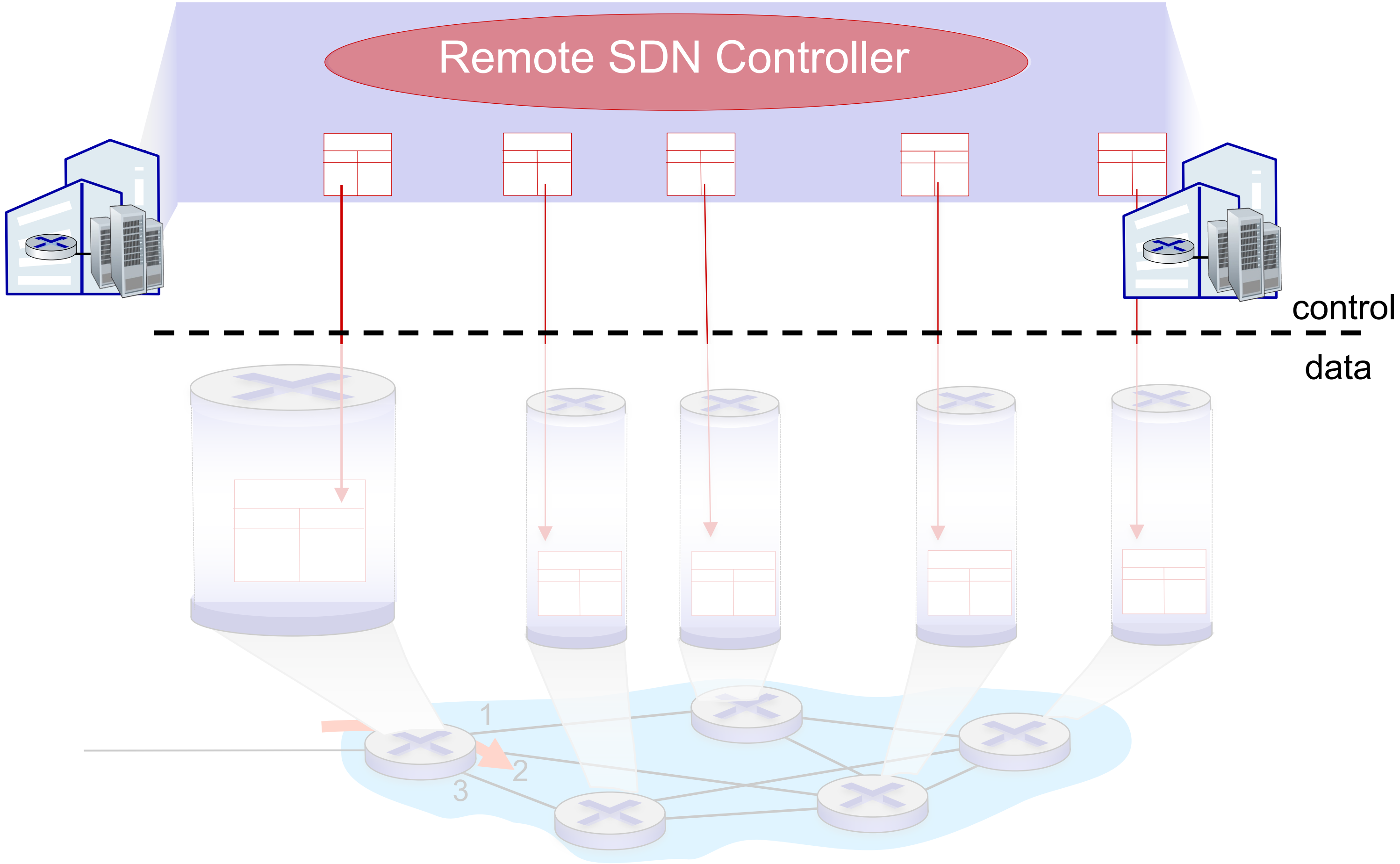


Local forwarding table

header	output
0100	3
0110	2
0111	2
1001	1

SDN: logically centralized abstraction:

- SDN's "logically-centralized" management approach provides *network-wide view*
- *logically*-centralized device, flow configuration and management
  - correctness: host, service reachability; loop freedom; isolation among virtual networks; access control policy

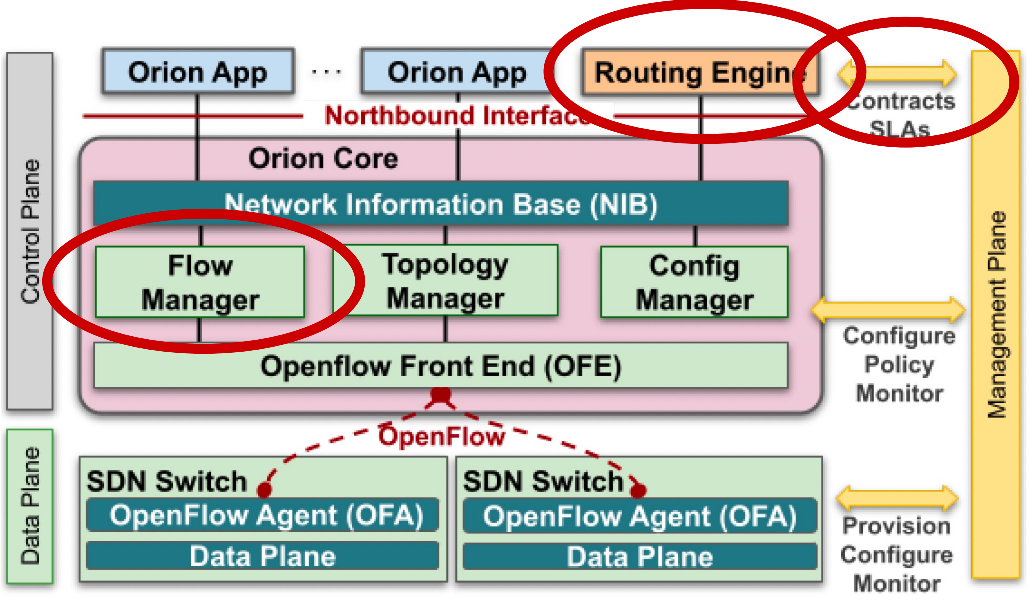


A migration away from “protocols” is underway

ORION: Google’s SDN control plane (*NSDI’21*): control plane for Google’s datacenter (Jupiter) and wide area (B4) networks

- routing (intradomain, iBGP), traffic engineering: implemented in *applications* on top of ORION core
- edge-edge flow-based controls (e.g., CoFlow scheduling) to meet contract SLAs
- management: pub-sub distributed microservices in Orion core, OpenFlow for switch signaling/monitoring

Orion SDN architecture and core apps



Note: ORION provides intradomain services within Google’s network

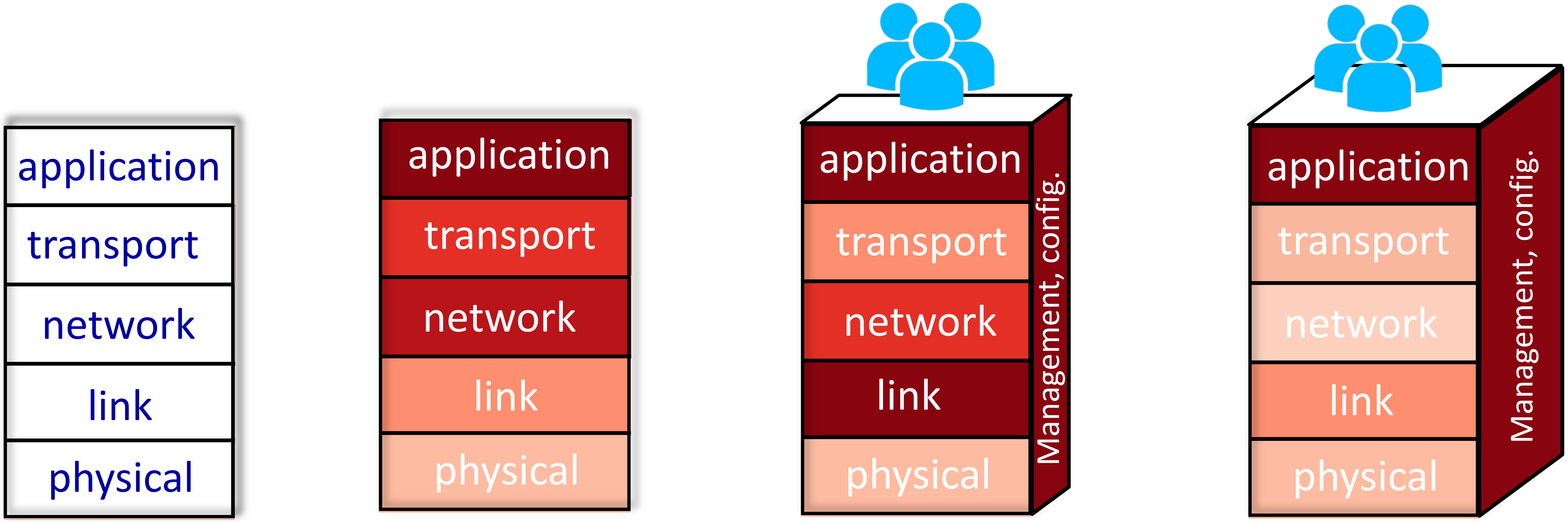
Are protocols “dead”?

Q: Will we still need protocols in 2030 ? Why?

A: Yes – protocols needed! But where?

- *internetworking* between different organizations, with separate control planes
  - eBGP, mobile packet cores in different organizations
- *application layer:* e.g., HTTP/3
- *identity management* (initial connection; capabilities, trust establishment)
  - 802.11: beaconing, DHCP, EAP
  - 4G/5G: MME-HLR/HSS DIAMETER over B6a interface

Evolution of interest in the protocol stack



application
transport
network
link
physical

application
transport
network
link
physical

application
transport
network
link
physical
Management, config.

application
transport
network
link
physical
Management, config.

pre-2000:
artifact,
performance

2000-2008:
distributed apps,
mobility
security

2008-2020:
data centers mobility
security, privacy
management

2020-2030:
mobility, IoT,
management, edge/core
cloud integration

Outline

Software: eating the world

Protocols

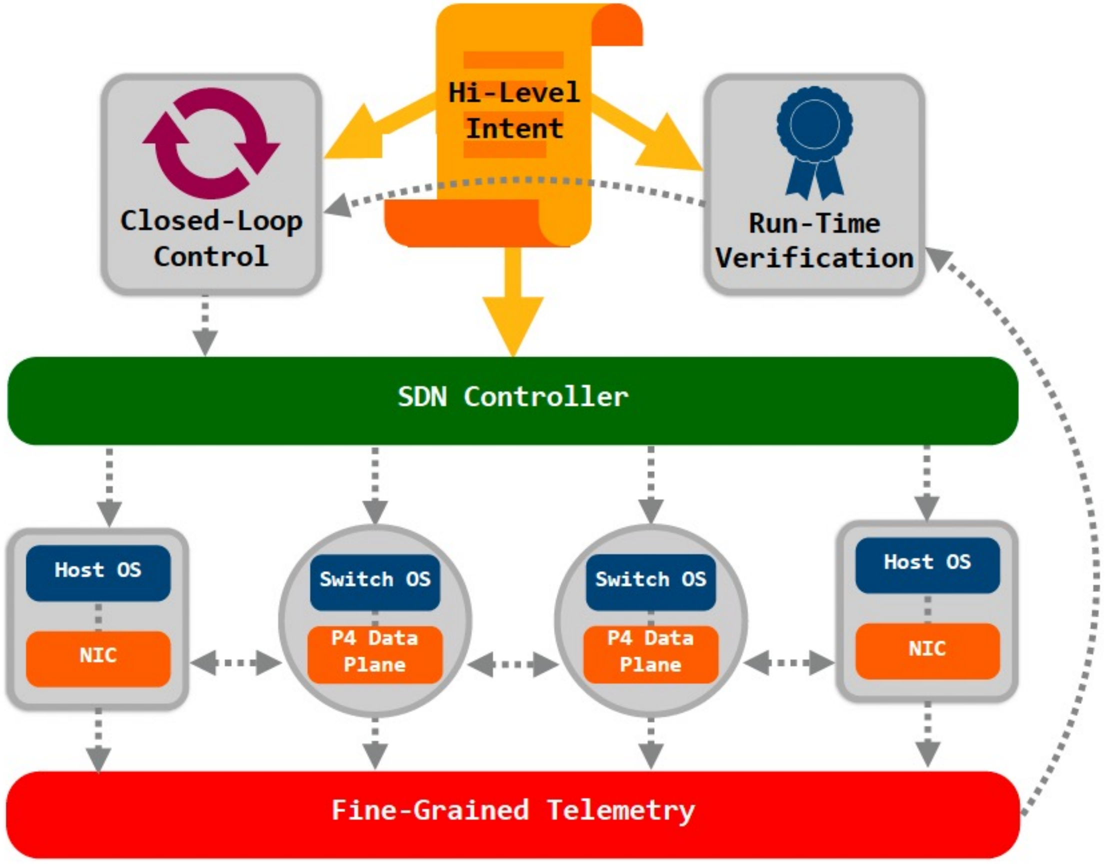
People

Pedagogy

Taking people
out
of the network
management loop

Intent-based networking

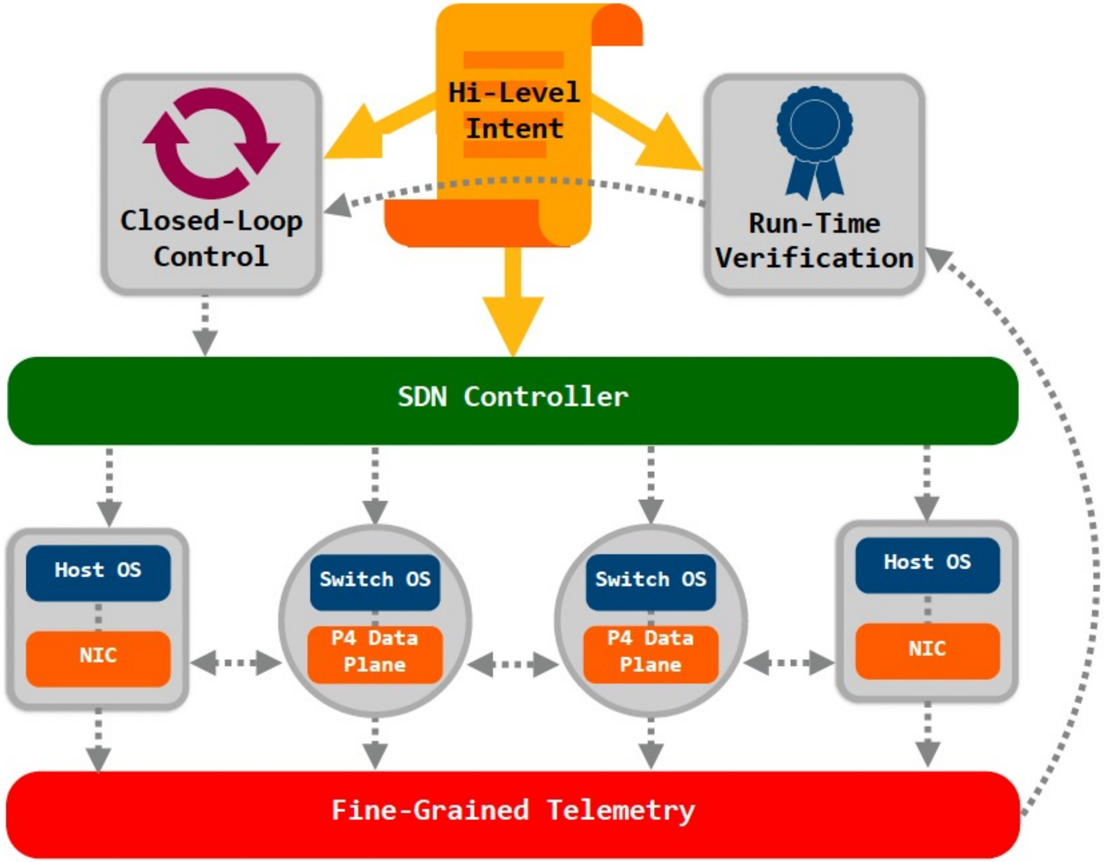
- specify management, design changes by "what" (the intended end state of the network) rather than "how"
- high level-intent doesn't change at rapid time scales
  - derived lower-level intents may change
  - lower-level intent and detailed configuration changes "compiled" (consistent with high-level intent)
  - automated, correctness, consistency checking



From "Using Deep Programmability to Put Network Owners in Control, Foster et al, ACM Computer Communication Review 2020.

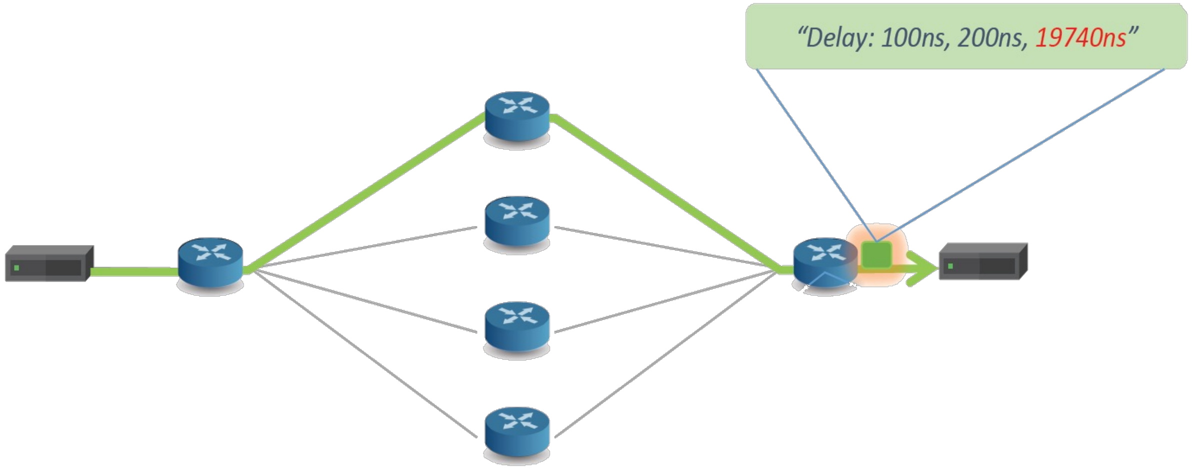
SDN, Intent-based networking: taking people *out* of control loops

- mostly automated "live" network management
  - no ssh/ CLI into routers
- network management "secret sauce" in higher level intent compilation, finer grain (automated) control loop apps



From "Using Deep Programmability to Put Network Owners in Control, Foster et al, ACM Computer Communication Review 2020.

Example: closed-loop monitoring, verification, healing



Verification transverse:

- performance SLAs specified using extension of temporal logic
- automatically generated, run-time P4-INT monitors: check progressively fine-grained performance measurements against SLA
  - e.g., from end-to-end to individual hops
- specified healing action taken when SLA violation detected

N. Choi *et al*., "Run-time Performance Monitoring, Verification, and Healing of End-to-End Services," *2019 IEEE Conference on Network Softwarization (NetSoft)*, Paris, France, 2019

Outline

Software: eating the world

Protocols

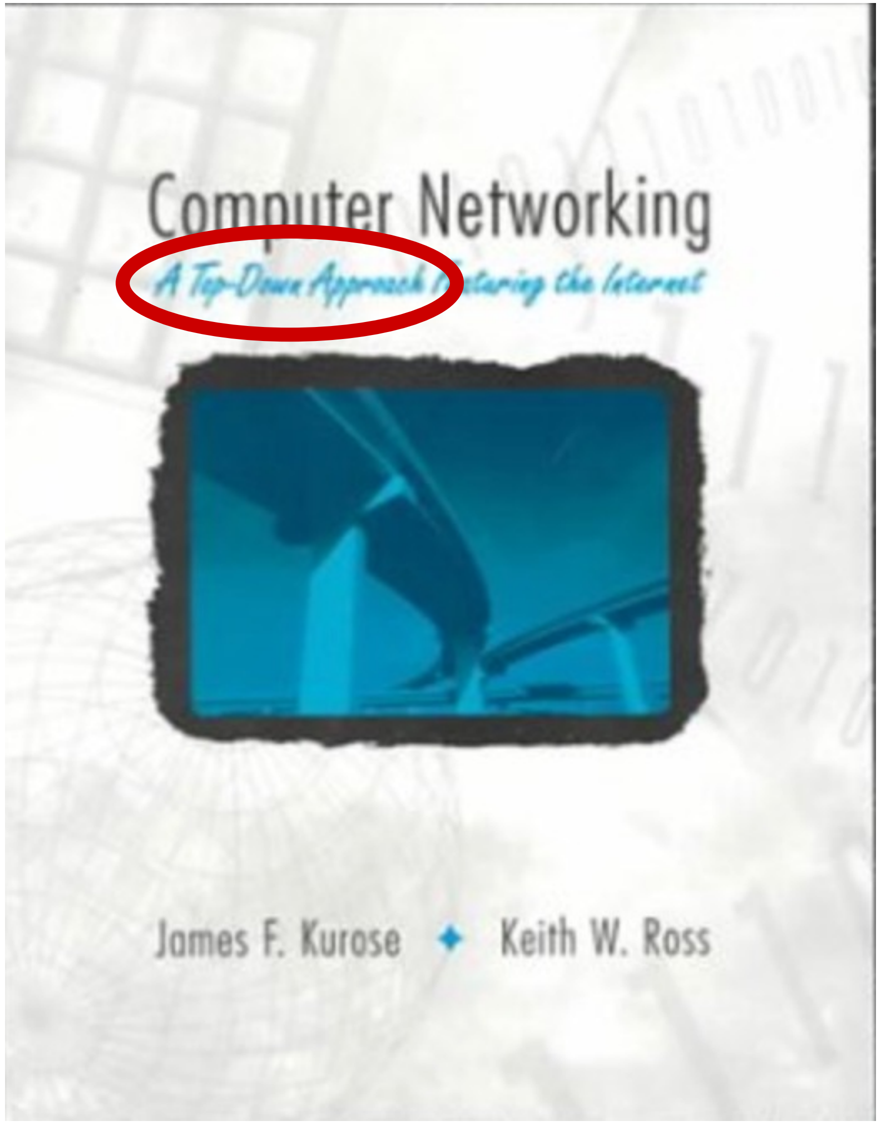
People

Pedagogy

What goes into an *introductory* computer networks course?

Teaching Intro Networking

- in late 1990's, a "top down" approach was novel and opposite of engineering "bottom up" approach
- reflected computer science's (top down) ease with building on top of abstractions, APIs, versus EE (bottom up) start with the basic building blocks and build up
    - operating systems had gone through similar evolution 10 years earlier

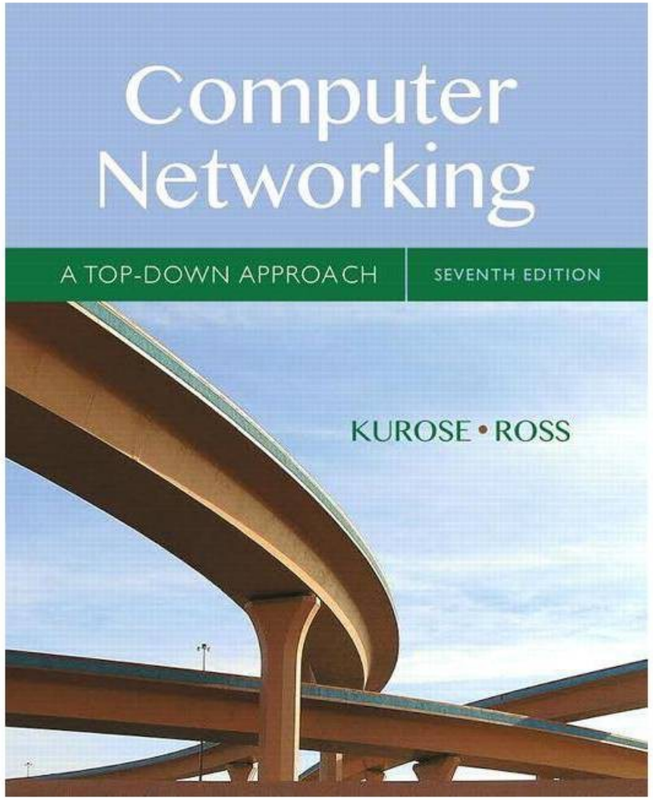

Computer Networking
A Top-Down Approach Featuring the Internet
James F. Kurose ✦ Keith W. Ross

Computer Networking: a Top-Down Approach Featuring the Internet, 1st edition, 1999.

Teaching Intro Networking (in a software-defined world)

2015: separation of network layer into separate control and data plane chapters

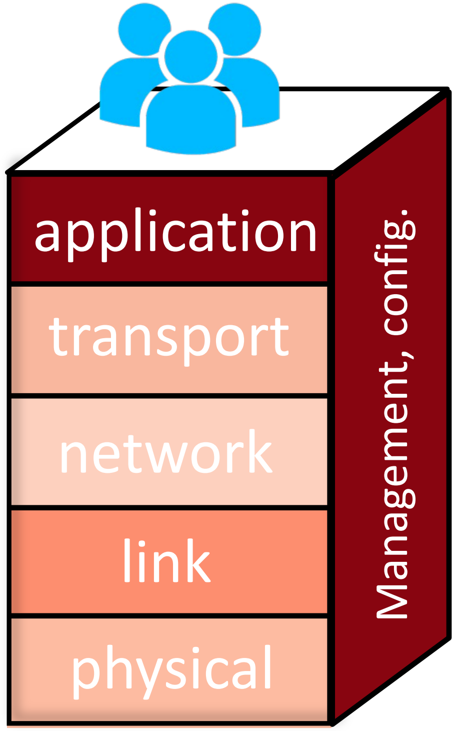
- data plane: generalized forwarding: match+action abstraction, Openflow
- control plane:
  - routing algorithms
  - implementation in routing protocols or SDN controllers
  - data models, NETCONF/YANG



Computer Networking: a Top-Down Approach, 7th edition, 2015.

Teaching Intro Networking (in a software-defined world)

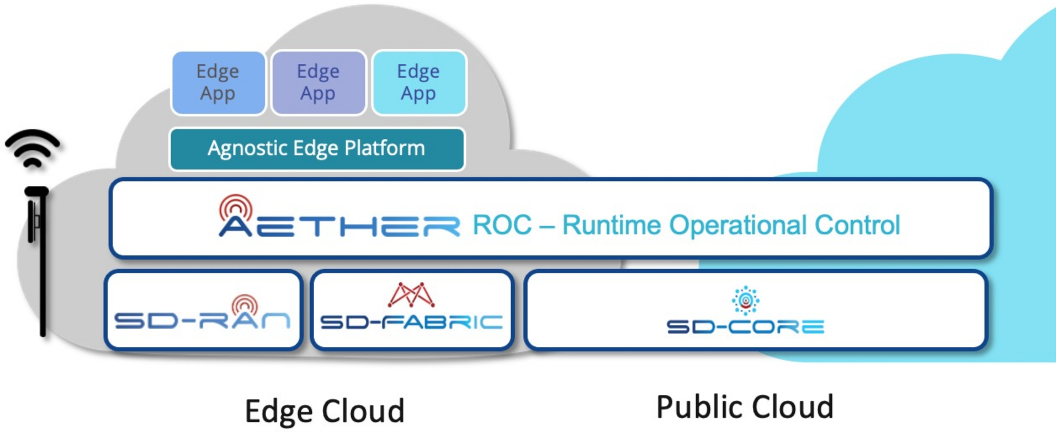
- protocols not quite "dead"
  - application, internetworking, management
- in the network, transport, link layers: from "distributed protocols" to "algorithms"
- distributed systems come to the fore:
  - distributed systems for scale, robustness
  - edge computing for real-time responsivity
  - Identity management
- What to teach?



2020-2030:
mobility, IoT,
management, edge/core
cloud integration

Closing thoughts: networking R&E

- Network "software-ization opens up opportunity to
  - develop/implement ideas *on top of* open, disaggregated software (e.g., O-RAN, AETHER, POWDER)
  - develop/implement ideas *in* open, disaggregated software

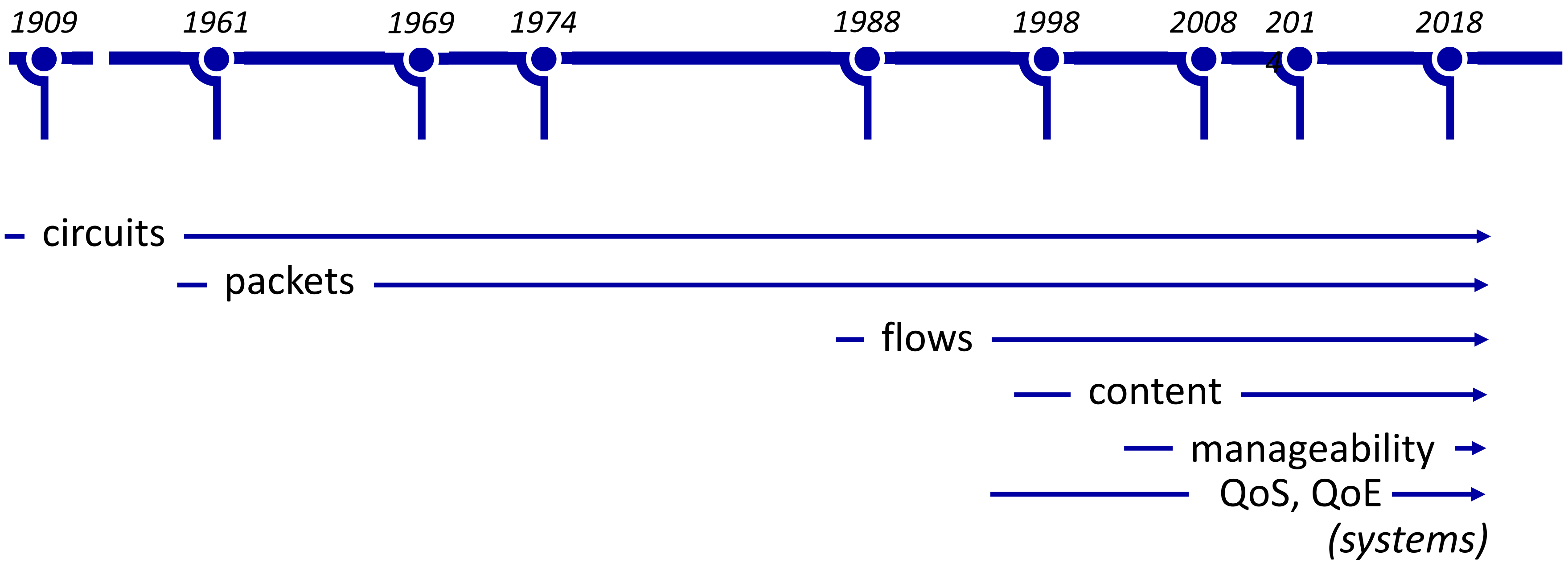


POWDER

Platforms for Advanced Wireless Research

Closing thoughts: networking R&E

- over time, changing network abstractions are becoming larger, more encompassing, more network-wide, more logically centralized, more *software-ized*



1909 *1961* *1969* *1974* *1988* *1998* *2008* *2014* *2018*

- circuits
- packets
- flows
- content
- manageability
- QoS, QoE

(systems)

THANKS!

Question or comments?

Software: eating the world

Protocols

People

Pedagogy

Are protocols dead?

Taking people *out* of the network management loop

What goes into an *introductory* computer networks course?