USING SOFTWARE ENGINEERING

TO TEACH NETWORKING

Pamela Zave

AT&T Labs—Research

Bedminster, New Jersey

Princeton University

Princeton, New Jersey

USA

IN 1997, THE INTERNET

WAS A WORLD-CHANGING PHENOMENON

WHAT HAS HAPPENED IN THE LAST TWENTY YEARS?

NEW CHALLENGES

- most of the world's . . .

  . . . telecommunication infrastructure
  . . . entertainment distribution . . .

  has moved to the Internet

- an explosion of security threats

- most networked devices are mobile

- cloud computing

- exhaustion of the IP address space

- the need for elastic resource allocation instead of over-provisioning

NEW IMPLEMENTATION TECHNOLOGIES

- have separated high-speed forwarding from control functions that can be implemented in software

- have made most network elements programmable

*as a result,
networks are now
software systems!*

AT THE SAME TIME, IN ACADEMIA . . .

NETWORKING IS AN IMPORTANT FIELD, BUT IT STRUGGLES TO BECOME A MATURE DISCIPLINE WITHIN COMPUTER SCIENCE

- core curriculum: teach how the Internet worked in 1997

  "In my college networking class I fell asleep at the start of the semester when the IP header was on the screen, and woke up at the end of the semester with the TCP header on the screen."

  this is as if databases had no relational model, . . . or today's curriculum in programming languages consisted of teaching Java

- theory concerns only resource allocation: queueing theory, control theory, linear and nonlinear optimization, algorithms

  these won't solve the problems of building secure software systems to meet an ever-expanding set of requirements

- the literature is full of narrow solutions to narrowly-defined problems

  there is little progress in generalizing the problems or solutions

  when challenged to propose "future Internet architectures," each team took one approach to a one-size-fits-all extreme

THE NETWORKING FIELD'S CONVENTIONAL WISDOM

"Our problems are due to the dominance of a single artifact, with its overwhelming size, complexity, and industrial investment."

"We must choose between working on short-term problems, or working on long-term research that *may* be difficult to apply."

and is certainly difficult to publish

"We are looking for the killer app for disruptive technology."

THE NETWORKING FIELD'S CONVENTIONAL WISDOM

"Our problems are due to the dominance of a single artifact, with its overwhelming size, complexity, and industrial investment."

"We must choose between working on short-term problems, or working on long-term research that *may* be difficult to apply."

and is certainly difficult to publish

"We are looking for the killer app for disruptive technology."

A CONTRARIAN VIEW

Despite drawing people from many backgrounds, the networking field lacks the crucial "gene" for appreciation of the importance of precise functional description.

without which there is no true abstraction or generalization

Essentially every paper has central terms that are ambiguous and not defined.

get used to a lot of shoulder-shrugging and "we know what we mean"

The biggest symptom is the core belief about the architecture of the Internet . . . .

BELIEF:

THIS IS A USEFUL AND ADEQUATE DESCRIPTION OF INTERNET ARCHITECTURE (WHICH IT WAS, IN 1997)

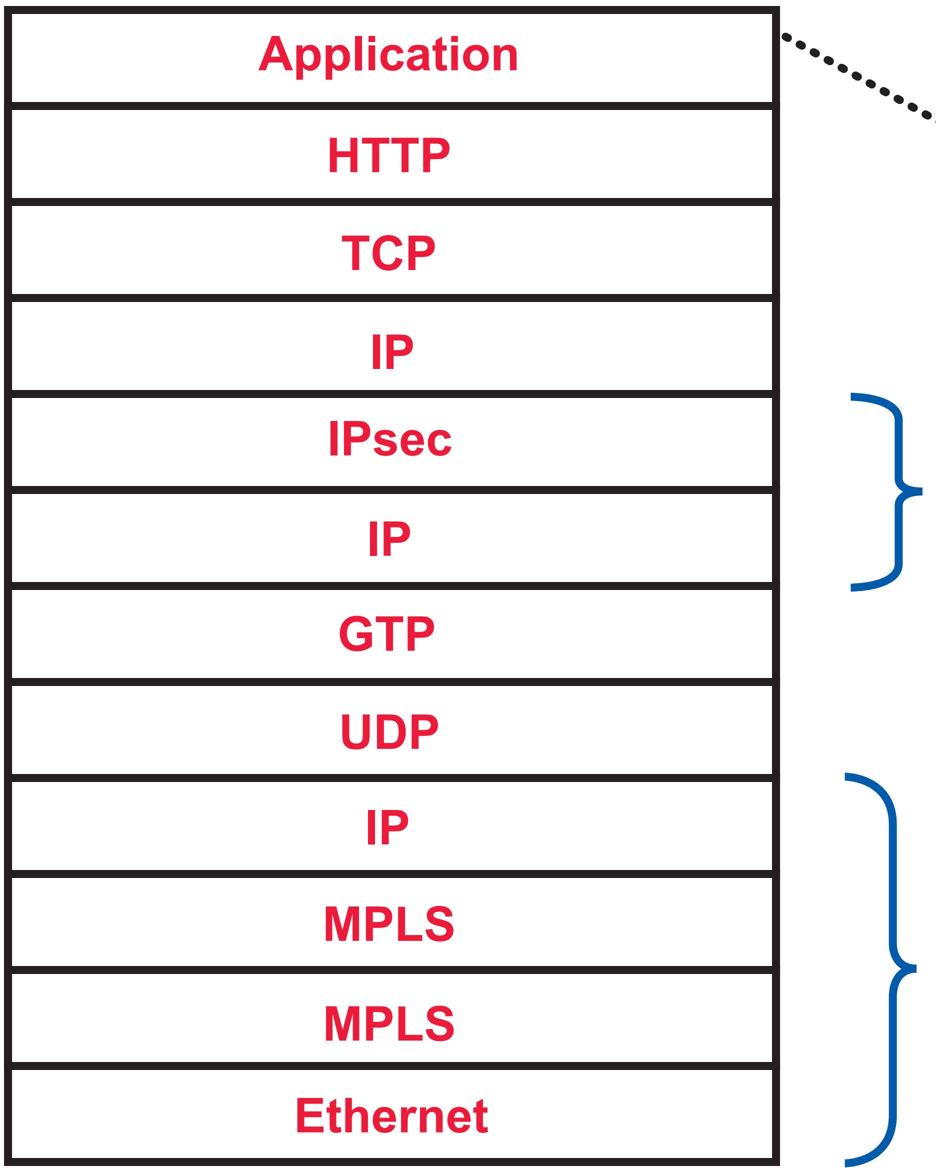
APPLICATION LAYER	applications and mnemonic names
TRANSPORT LAYER	reliable byte streams, messages
NETWORK LAYER	best-effort global packet delivery
LINK LAYER	best-effort local packet delivery
PHYSICAL LAYER	many physical media (wires, optical fibers, radio channels)

and so we expect a typical packet to look like this

HTTP header
TCP header
IP header
Ethernet header

THE REALITY: THIS IS A TYPICAL PACKET IN THE AT&T BACKBONE

packets sampled elsewhere would look different, but might be equally complex



12 headers instead of 4, with 3 IP headers!

security

cellular service (mobility, QoS, billing)

15 + load-balancing algorithms operate on this packet, most of them understood and tested only in isolation

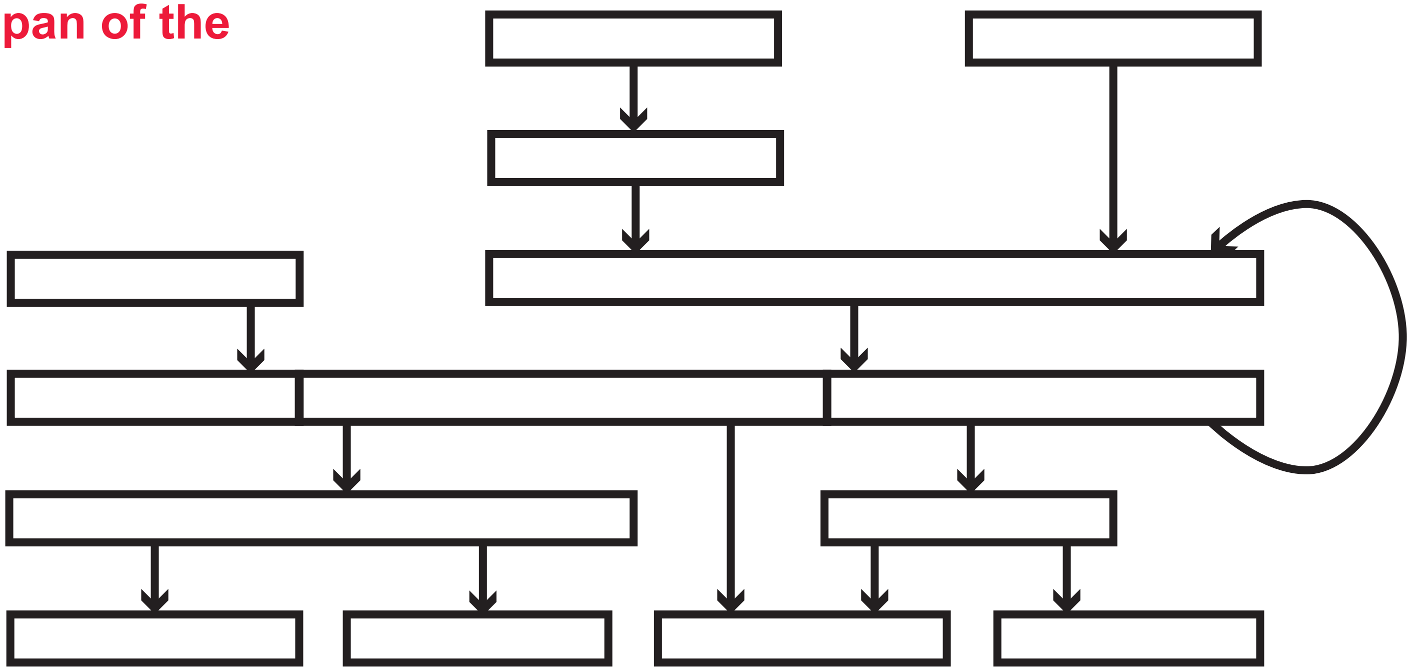
multiple layers of resource management

THE INTERNET IS ACTUALLY A COMPOSITION OF MANY NETWORKS

each network has all the basic mechanisms, . . .

. . . but in each network they are specialized for the particular purpose and span of the network

because all networks have fundamental similarity, they can have common interfaces for composition



TCP/UDP/IP is just the common software that most networked devices have installed

this structure is obvious from observation, and it makes sense—how else could we get the flexibility to satisfy an ever-expanding roster of requirements and stakeholders?

THE FIELD OF NETWORKING NEEDS A THEORY OF COMPOSITIONAL NETWORK ARCHITECTURE

WHY? BECAUSE THIS IS WHAT IS NEEDED TO . . .

- **understand networks as software systems with ever-expanding requirements**

  introducing modularity is what we software engineers understand best

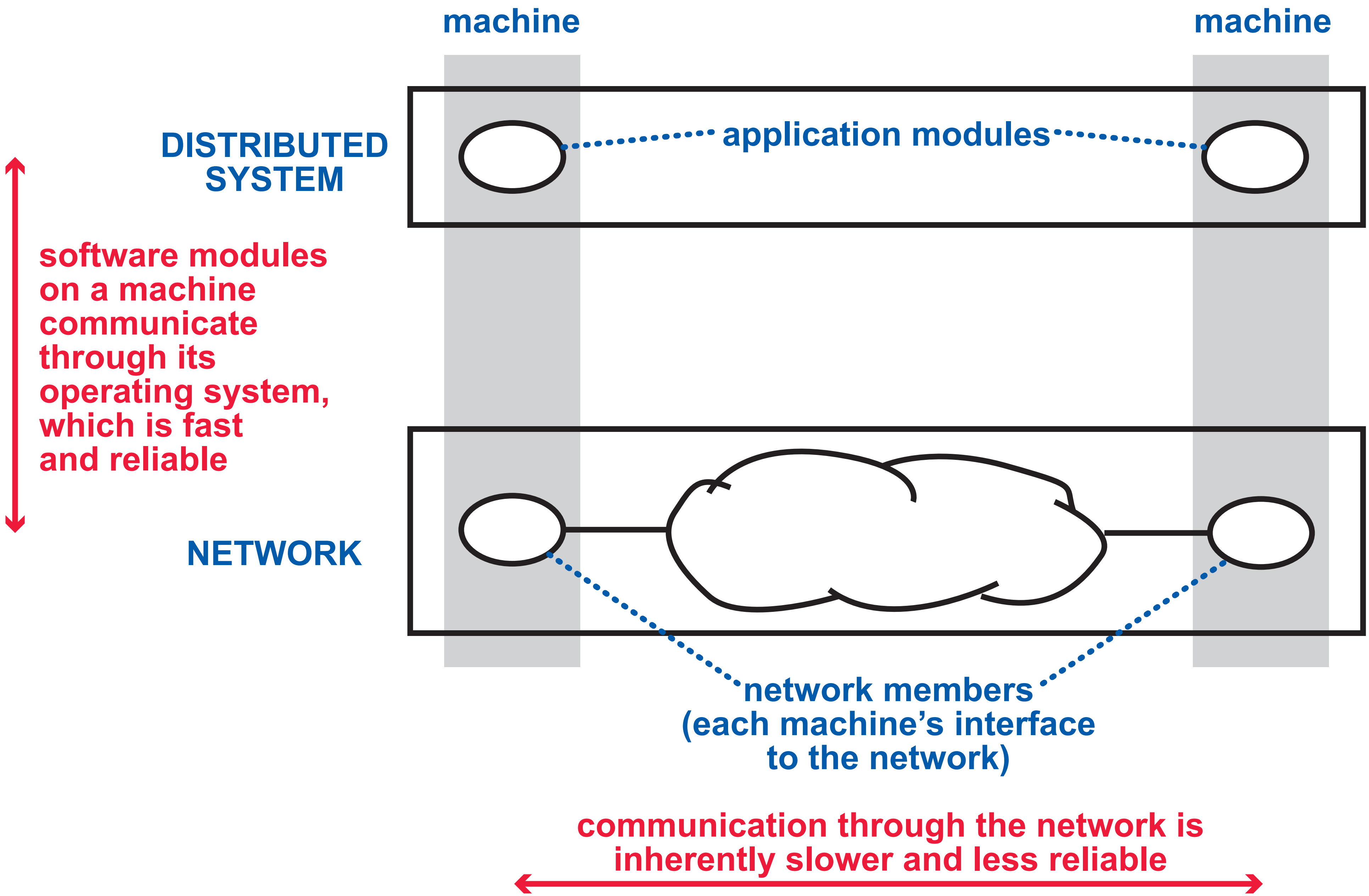
- **jump-start a whole new body of theory about the functions of network software**

- **show networking researchers that the Internet is already far more flexible than they think it is**

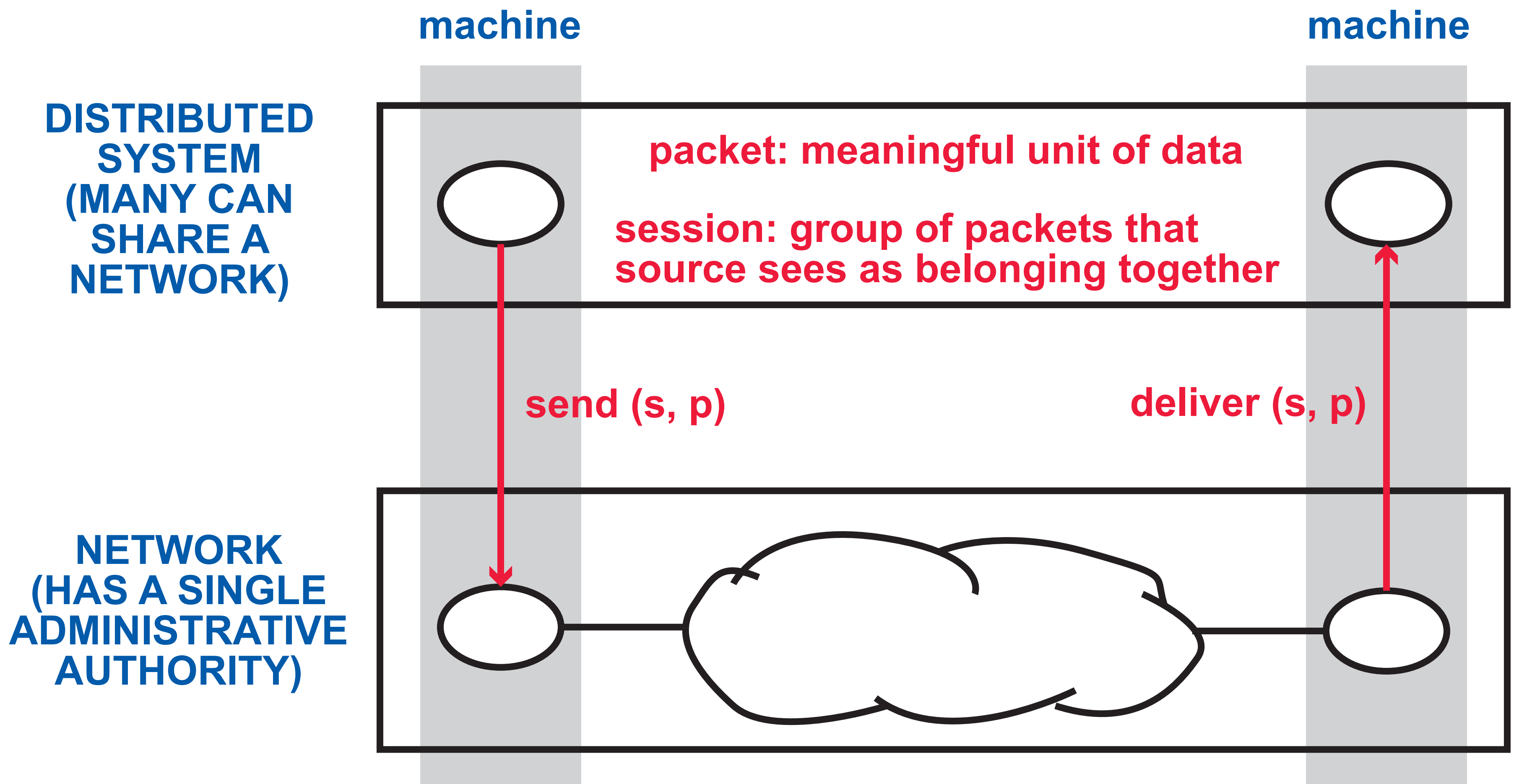
- **to spread knowledge of networking beyond the current guild of people who have devoted themselves to arcane details**

  already, researchers in programming languages are jumping at the opportunities offered by the increased programmability of networks

NETWORKS SUPPORT DISTRIBUTED SYSTEMS BY PROVIDING THEM WITH COMMUNICATION SERVICES



REQUIREMENTS ON SESSIONS



REACHABILITY
- what are the possible destinations?

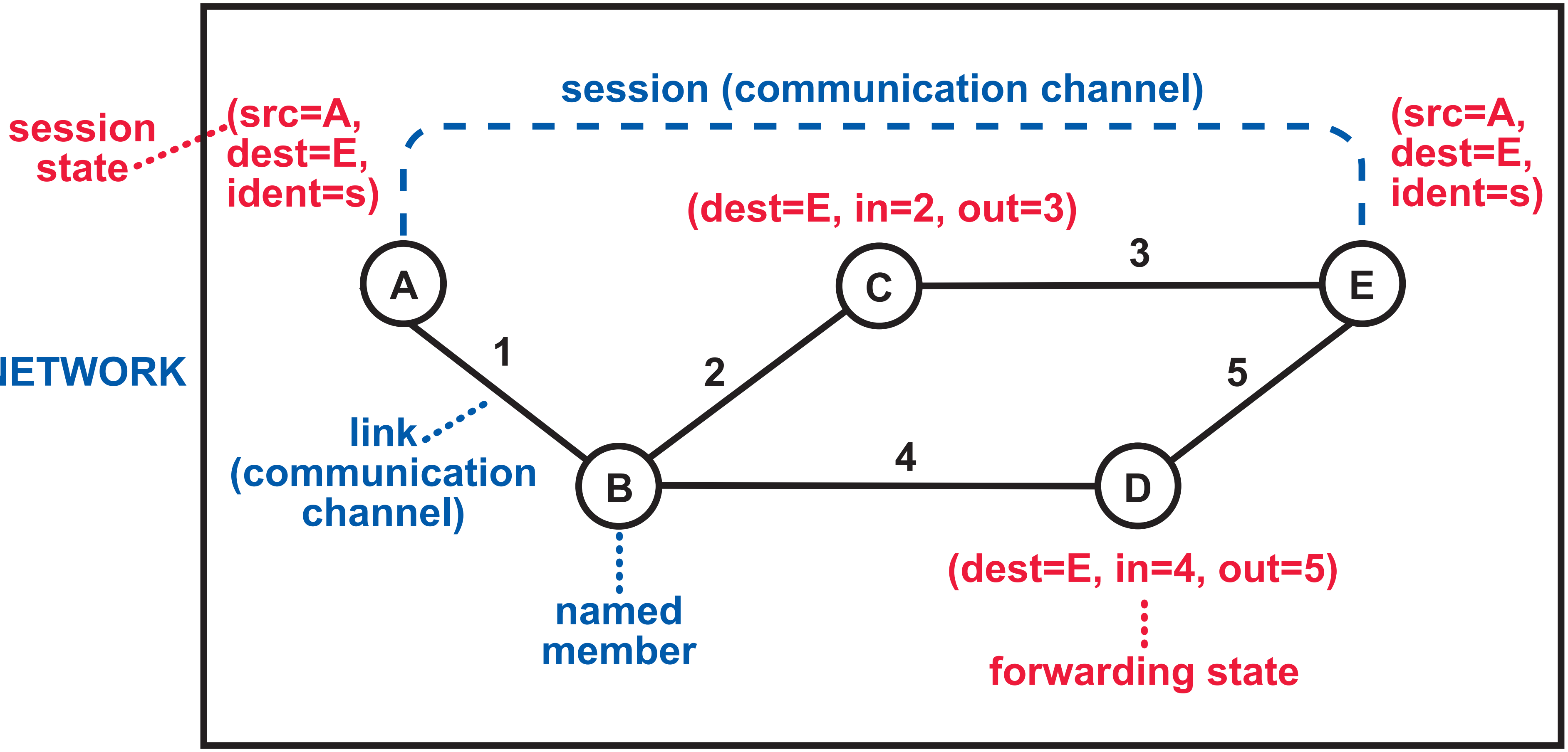
PERFORMANCE
- minimum bandwidth
- maximum latency

RELIABILITY
- packet loss

SYNCHRONIZATION
- systems use network communication for this as well as data transfer

SECURITY
- DoS protection
- malware protection
- authentication
- privacy
- data integrity
- lawful intercept

PARTS AND STATE OF A NETWORK



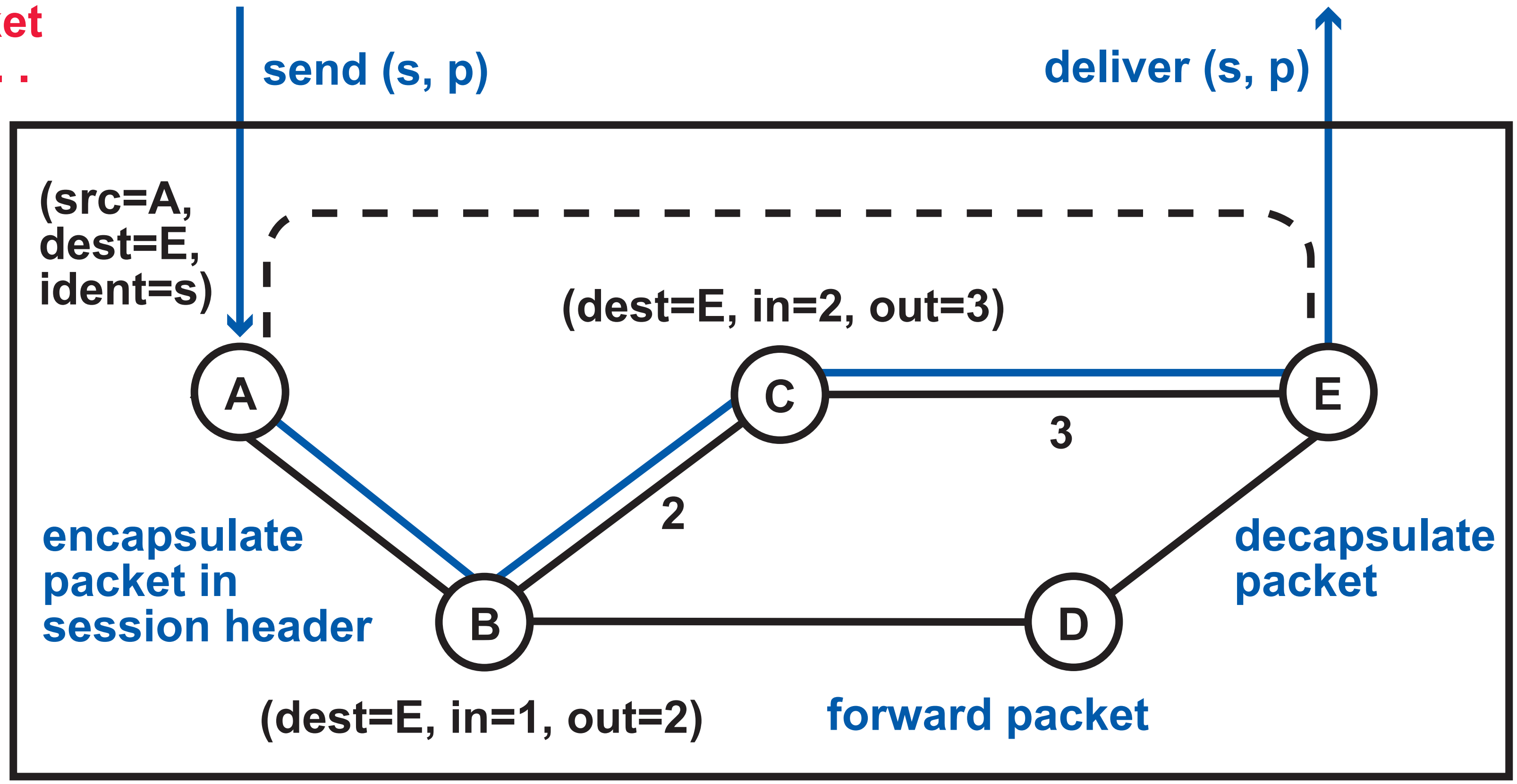
Some parts and state components are created "on demand", which requires additional user interfaces.

BEHAVIOR OF A NETWORK

the "DATA PLANE" does the packet processing, . . .

. . . also creates on-demand state, . . .

. . . and satisfies other session requirements through session protocols and middleboxes



packet-processing, on-demand state, and TRUST BOUNDARIES are modeled

the "CONTROL PLANE" maintains the parts and state components that are not on-demand—usually includes the traditional performance monitoring and routing

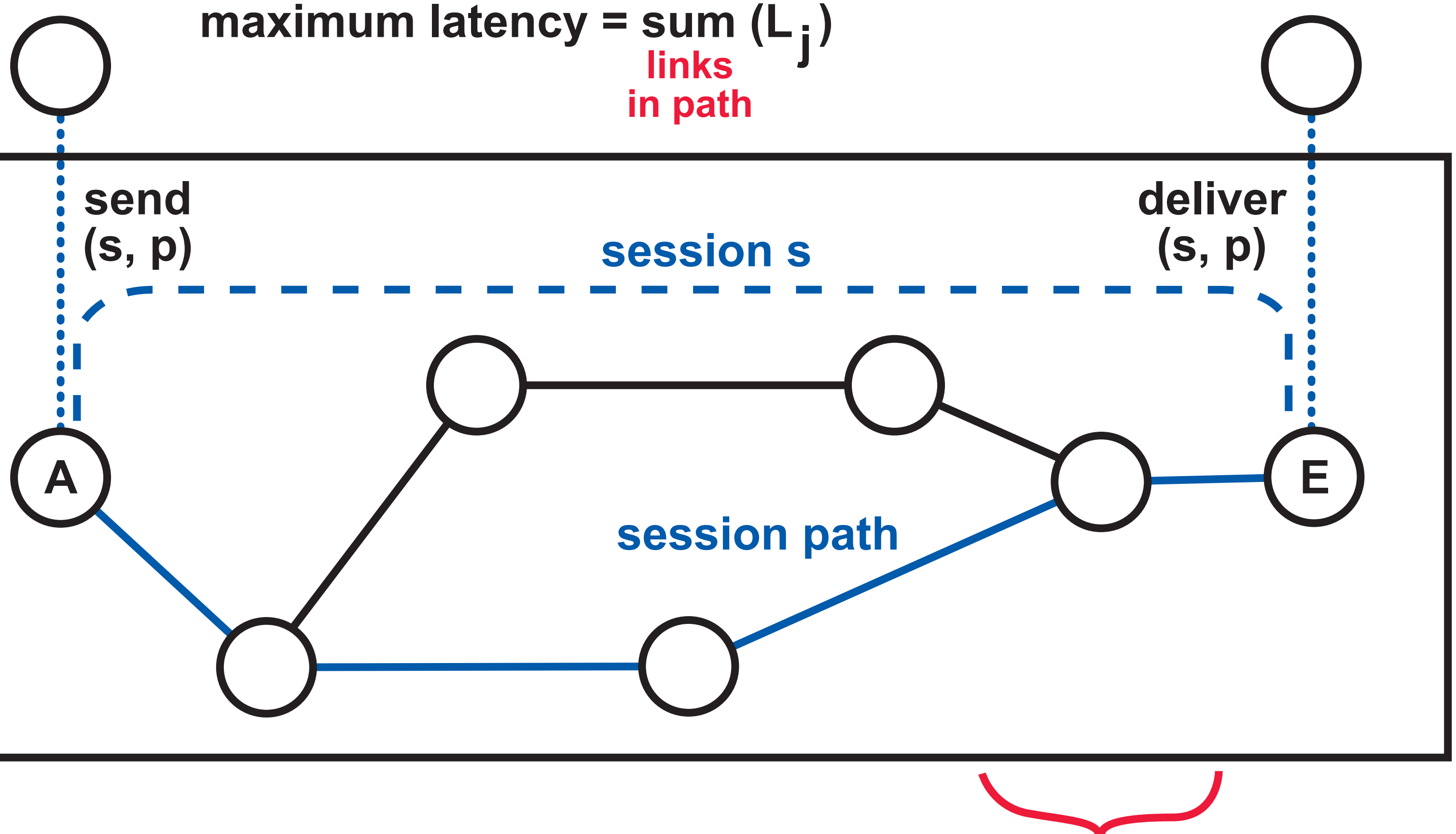
we need to formalize enough for composition and reasoning about requirements, but not too much

SELF-CONTAINED REASONING ABOUT A NETWORK

SESSION PERFORMANCE

minimum bandwidth = $\min_{\text{links in path}} (S_j(B_j))$ — S_j: session's share of bandwidth

maximum latency = $\text{sum}_{\text{links in path}} (L_j)$



REACHABILITY

forwarding relation says what can be reached from A in one hop

reachability from A is the transitive closure of the forwarding relation

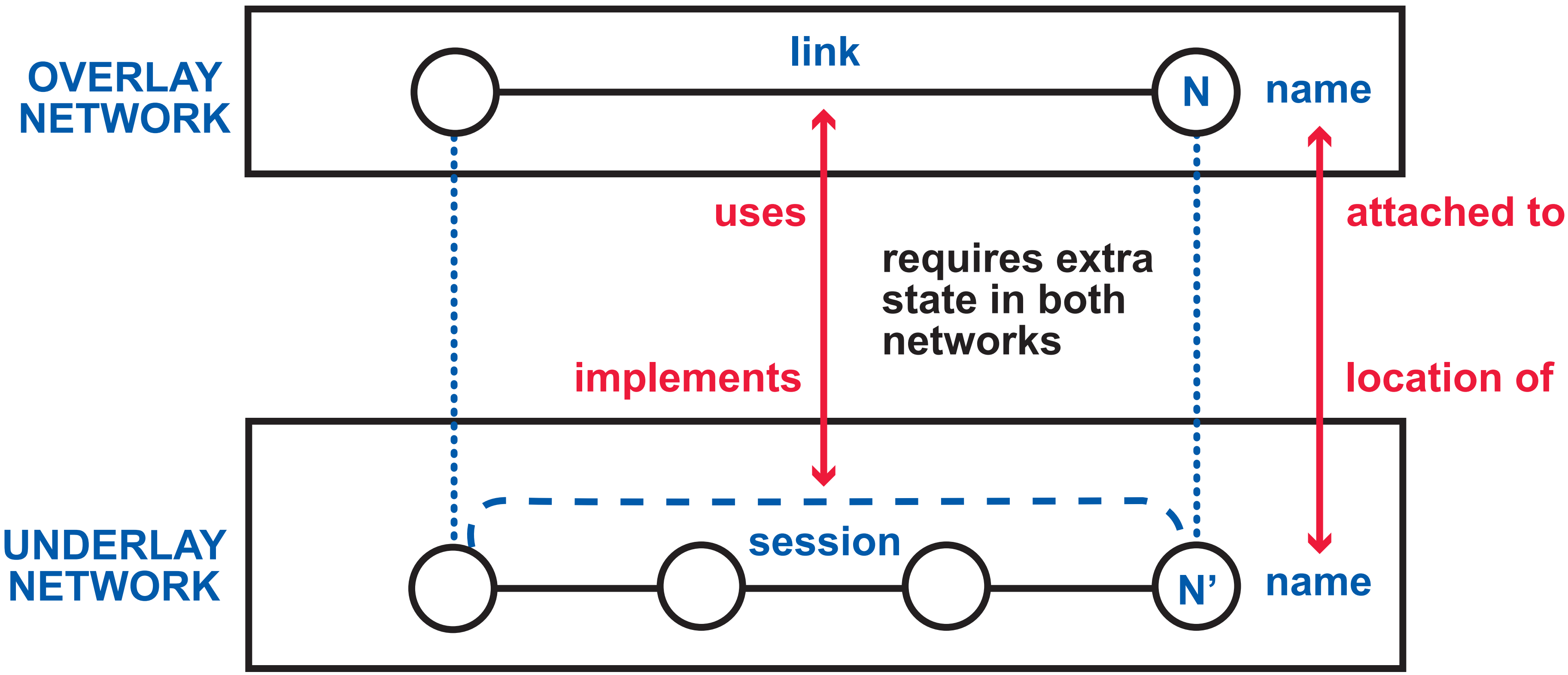
SECURITY

all paths to E go through middleboxes that protect it from DoS attacks and malware

A COMPOSITION OPERATOR: LAYERING

A link in an "overlay" network . . .

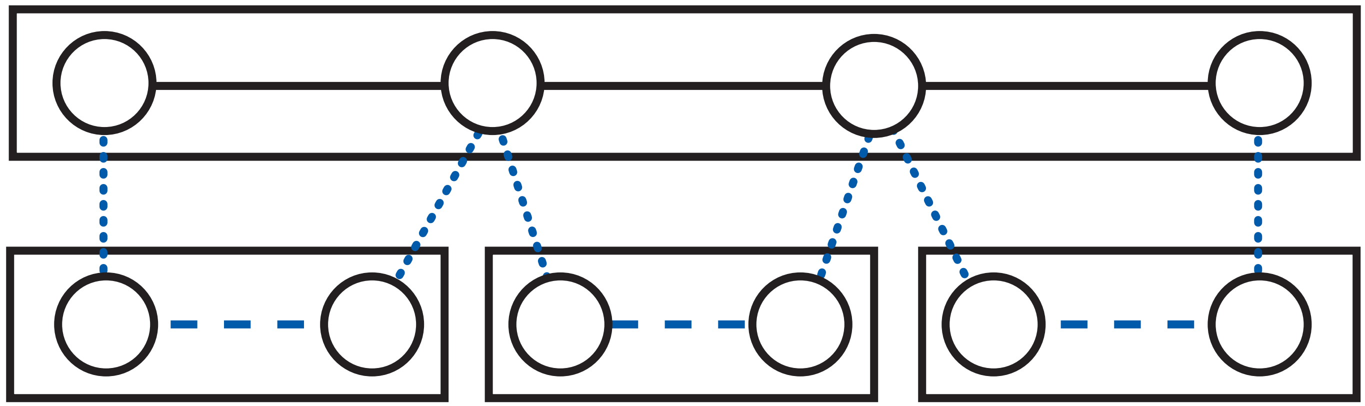
. . . is implemented by a session in an "underlay" network.

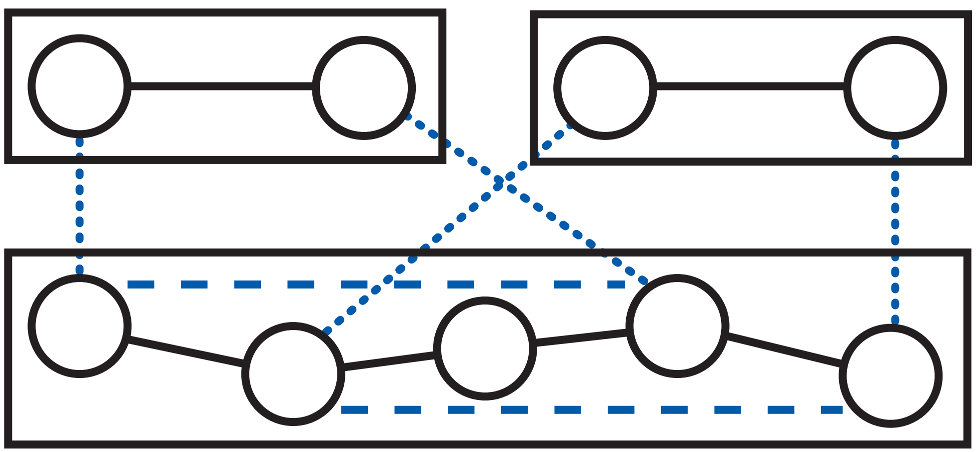


Compositional reasoning requires nothing new—the specified properties of the underlay session are simply the assumed properties of the overlay link.

LAYERING HAS MANY USES

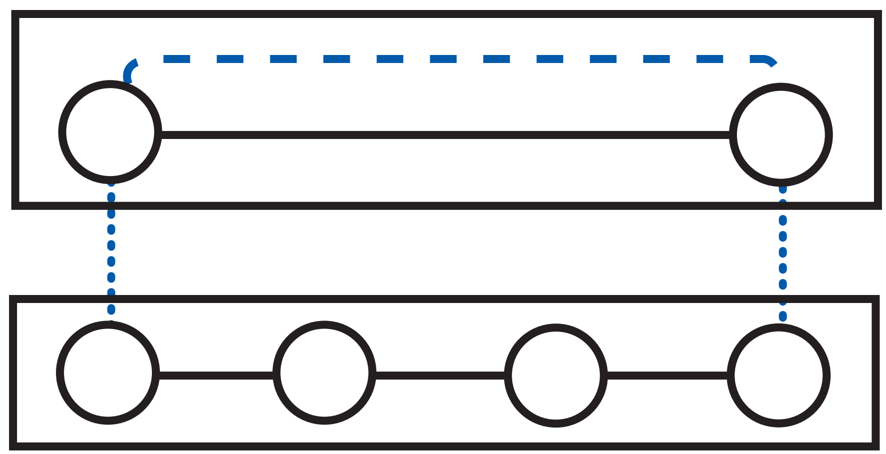
to build a network with a larger span out of smaller, heterogeneous networks





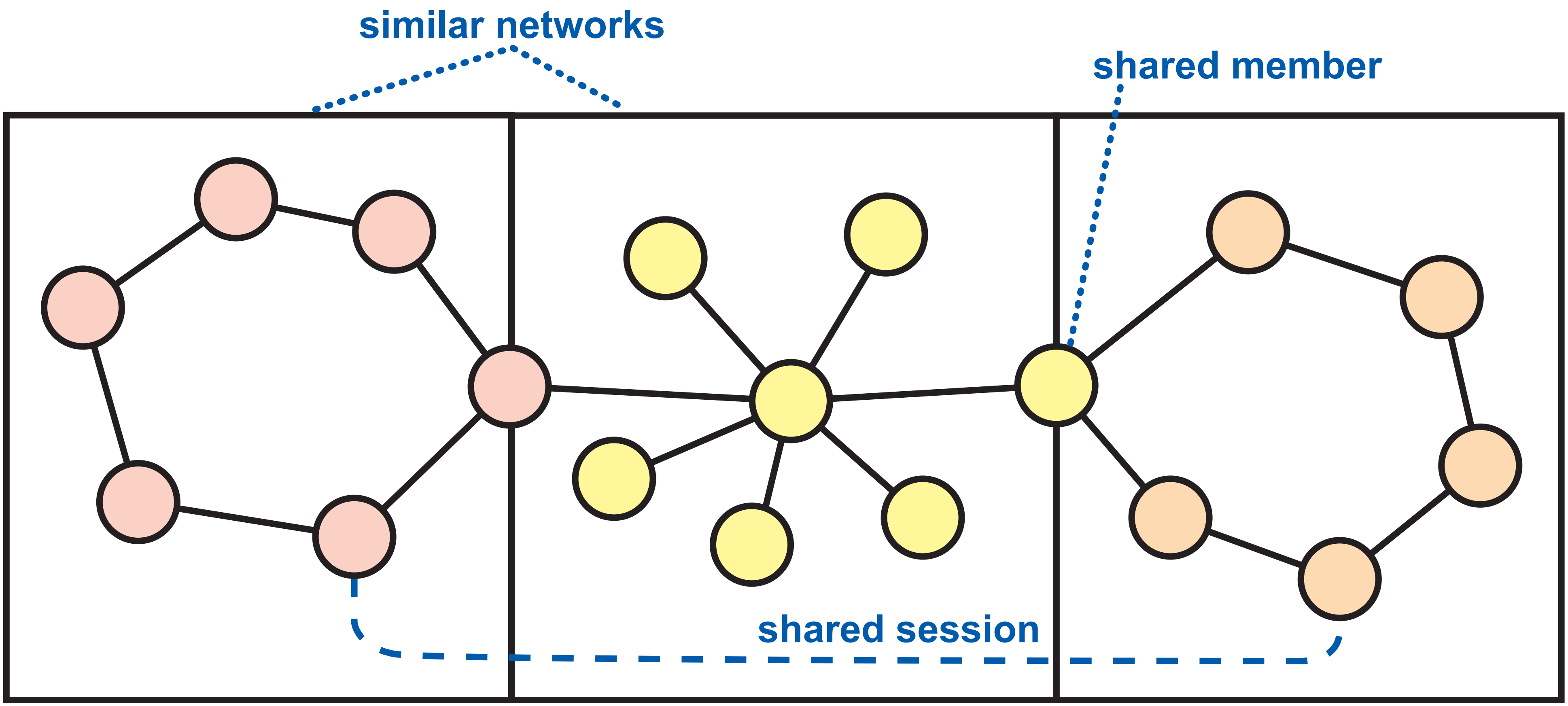
to share the resources of a network in a disciplined way

to build improved communication services on top of an existing network



A COMPOSITION OPERATOR: BRIDGING

BRIDGING EXTENDS THE REACH OF SIMILAR NETWORKS



because each network is autonomous, a shared member is usually owned and trusted by one network, not the other

THEORY CONTENTS

A FORMAL MODEL OF A NETWORK

- customizable with properties and libraries
- composable
- compositions of networks can be verified or simulated

VALIDATED DEFINITIONS OF PROPERTIES

- requirements
- consistency properties
- design properties (specializations)

CHANGE ANALYSIS

- what sequences of changes can the control plane make while preserving consistency and other properties throughout?

THEOREMS

- theorems relate the properties of networks (or compositions of them) to each other
- a sufficiently general theorem is called a “principle”

THEORY CONTENTS

A FORMAL MODEL OF A NETWORK

- customizable with properties and libraries
- composable
- compositions of networks can be verified or simulated

VALIDATED DEFINITIONS OF PROPERTIES

- requirements
- consistency properties
- design properties (specializations)

CHANGE ANALYSIS

- what sequences of changes can the control plane make while preserving consistency and other properties throughout?

THEOREMS

- theorems relate the properties of networks (or compositions of them) to each other
- a sufficiently general theorem is called a "principle"

THEORY USES

UNDERSTAND . . .

- how to satisfy requirements
- structured trade-off spaces
- find more solution patterns
- make precise comparisons

GENERALIZE, RE-USE, OPTIMIZE, GENERATE, VERIFY . . .

- data plane software and hardware
- eventually, the control plane

TEACH AND LEARN . . .

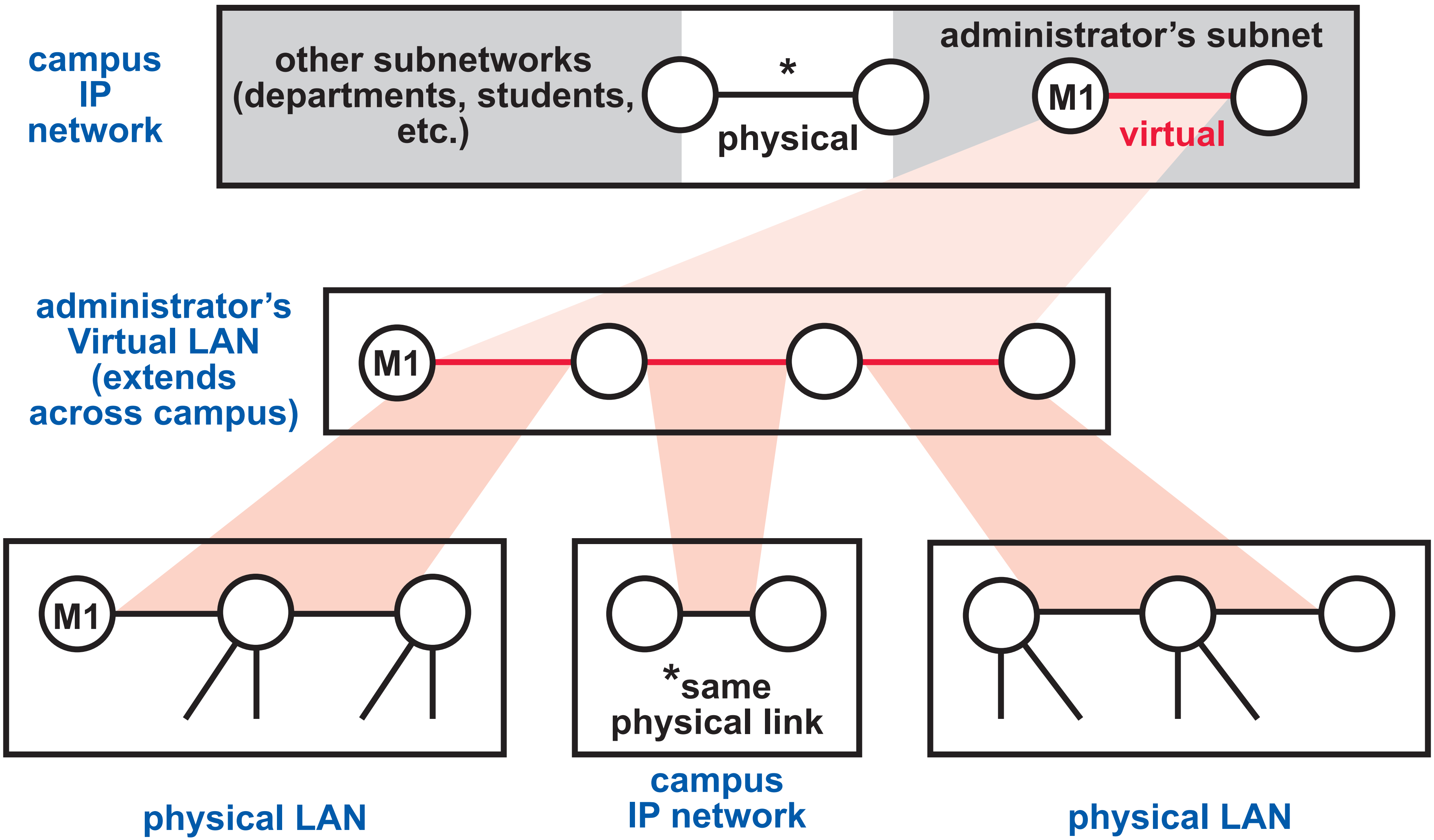
- help people understand networking more quickly and more deeply . . .

  . . . by teaching principles rather than details

DEFINING CONSISTENCY

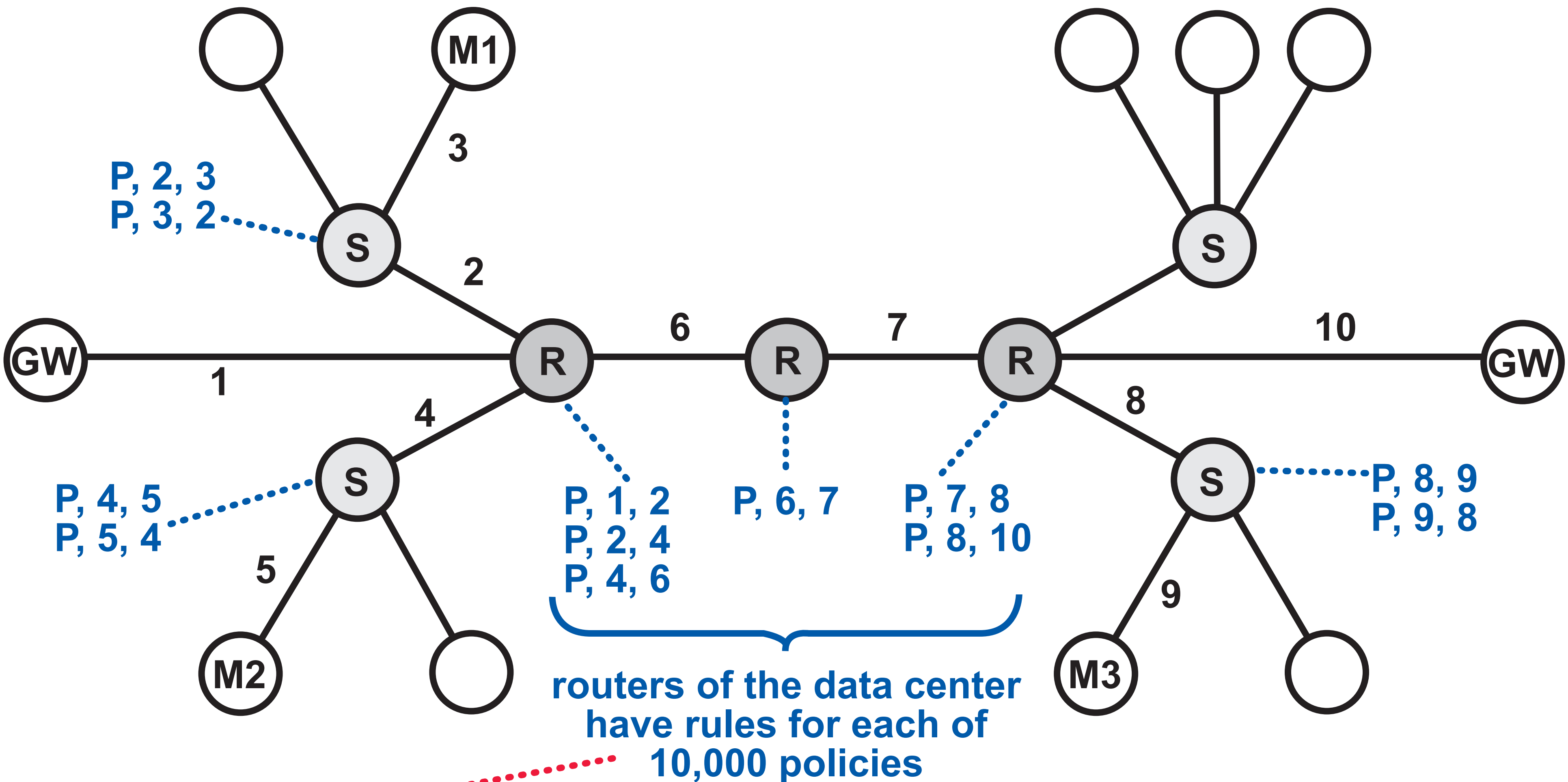
there cannot be cycles in resource usage,
. . . but this applies to links,
. . . not to networks

EXAMPLE: a campus (private) IP network, with a "VXLAN" architecture



REASONING ABOUT COSTS 1

POLICY: packets that match pattern P must go through middleboxes of types <M1, M2, M3>

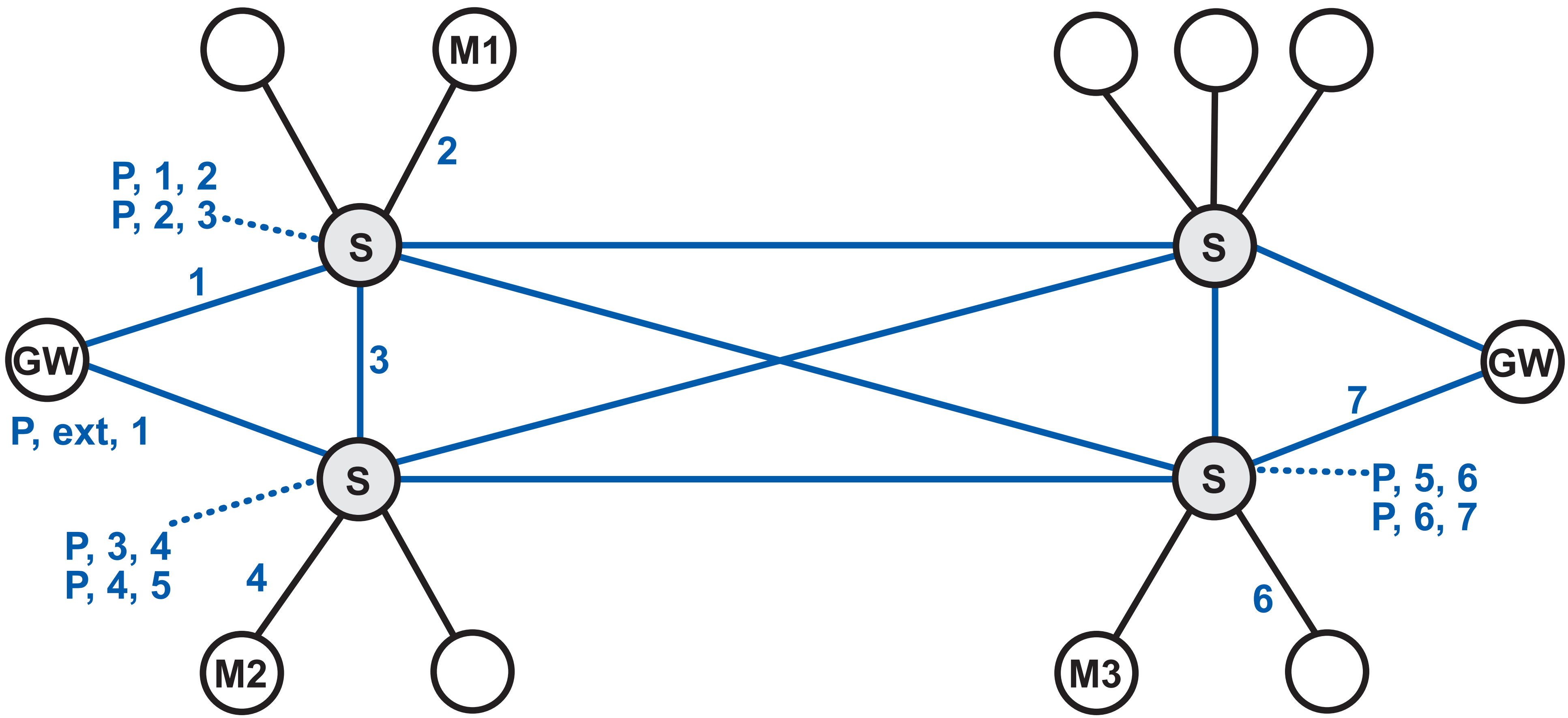


not possible!
. . . and the rules would change for . . .

- topology changes
- policy changes
- node and link failures
- fluctuations in load

REASONING ABOUT COSTS 2

because paths in the previous network are completely determined by switches, a general theorem says that this network is equivalent



the inter-switch links are implemented by an underlay network with the centralized routers, and only enough forwarding paths to connect the switches

each cause for change affects one network only

THE INTERNET OF THINGS

according to short-term estimates, . . .
there will be 25 times as many networked things as cellphones, . . .
and they will need mobile connectivity at 1/25 the cost

the **FIRST RESULT** of the theory of
Compositional Network Architecture
was that there are two patterns for implementing mobility:

DYNAMIC ROUTING MOBILITY

- built into network infrastructure, changes routing as devices move
- very expensive on a large scale
- this is what cellular providers use

SESSION-LOCATION MOBILITY

- uses the session protocol to transmit new endpoint locations
- easy to implement on a large scale
- security and deployment problems

INTERNET OF THINGS: RESEARCH CHALLENGE

Use Compositional Network Architecture to find a version of mobility that is scalable, secure, and easily deployed.

SECURITY

- use the model (isolation, trust boundaries) to limit where security is needed
- provide provable security where it is needed

DEPLOYMENT

- design robust interoperation with the existing Internet
- select appropriate technology for distributed directories

PROTOCOLS

- design protocols to minimize the burden on low-power devices, without sacrificing other requirements

THIS CHALLENGE REQUIRES:

- architectural flexibility
- rigorous reasoning

exactly what the theory provides!

TEACHING

my graduate course "Patterns in Network Architecture" at Princeton showed how all the new Internet features since 1997 can be explained and modeled with compositions of networks

including cloud computing, data-centric networking, multicast, multihoming, and proxies

WERE ANY GENES TRANSPLANTED?

- **it took most of the semester to get across that I was using terms with mathematical precision, not in the usual handwaving way**

- **I think they really learned something about seeing the big picture**

- **to learn a lot of specifics, they would need a more competent professor**

TEACHING

my graduate course "Patterns in Network Architecture" at Princeton showed how all the new Internet features since 1997 can be explained and modeled with compositions of networks

including cloud computing, data-centric networking, multicast, multihoming, and proxies

WERE ANY GENES TRANSPLANTED?

- it took most of the semester to get across that I was using terms with mathematical precision, not in the usual handwaving way
- I think they really learned something about seeing the big picture
- to learn a lot of specifics, they would need a more competent professor

THIS IS ONLY THE BEGINNING!

- continuing to develop the theory
- there is a planned application of the theory at AT&T, for data plane implementation
- continuing to improve the course

experienced researchers in other fields could learn the important things about networking very quickly and efficiently

ACKNOWLEDGMENTS

This talk is based on joint work with Jennifer Rexford of Princeton University.

John Day's book *Patterns in Network Architecture* was my Rosetta Stone.

All the formal modeling has been done in Alloy, with thanks to Daniel Jackson of MIT for making it possible.

`www.cs.princeton.edu/courses/archive/spr17/cos598D`