

# **Communication Networks: Technology & Protocols**



Stavros Tripakis (stavros@eecs)

**Lecture 3**  
**August 27**

# Logistics



- Web site:
  - [www.cs.berkeley.edu/~amc/eecs122](http://www.cs.berkeley.edu/~amc/eecs122)
- Book:
  - Jean Walrand, *Communication Networks, A first course*, **2nd edition**, 1998
- Enrollment:
  - Please check your name on the class list or waiting list (add your name if you are not there already).
- Office hours: Mon 10-11am, Thu 4-5pm
- Grading: 25% + 2\*15% + 30% + [15,30]%

# Summary of last lecture



- Networks provide **services**.
- Networks have to be **open** (interoperable, extensible).
- Networks have to be **scalable**.
- Networks have to support **diverse** applications.
- Designers have to make **good utilization** of the resources of the network (in order to achieve **cost-effectiveness**).

Note: diversity often in conflict with cost effectiveness.

# Summary: applications



- Different applications have **different** (bit-rate) **requirements**:
  - Constant bit-rate traffic (e.g., voice, video).
  - Bursty traffic (e.g., file transfer, web, e-mail).
- Other requirements: delay, loss probability.
- Quality of service (QOS).

# Summary: Telephone network & Internet

- Telephone network: circuit-switched.
  - Fixed allocation of resources.
  - Well-suited for CBR traffic, inefficient for bursty traffic.
  - Interoperable and scalable, but not easy to extend.
- Internet: packet-switched.
  - Allocation of resources on-demand.
  - Interoperable, scalable, diverse, very efficient.
  - But: no QOS guarantees.

# **The Internet: properties.**



- Interoperability : good.
- Scalability: good (IP addresses ?).
- Diversity / Extensibility : very high, but no guarantees for applications.
- Cost-effectiveness : very good.

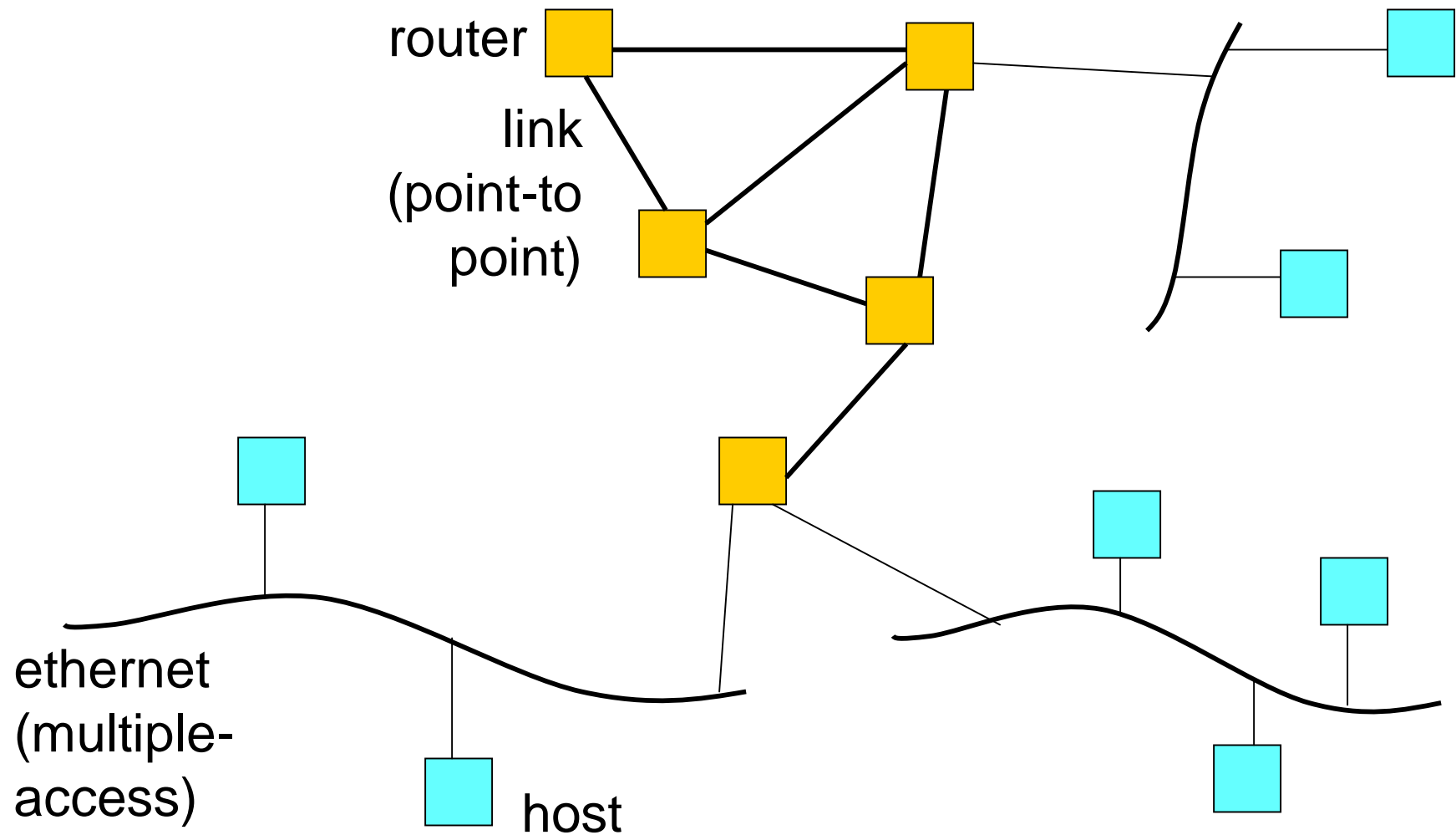
# **Broadband Integrated Services Data Network (BISDN)**



Can we extend the Internet so that it supports all kinds of applications ?

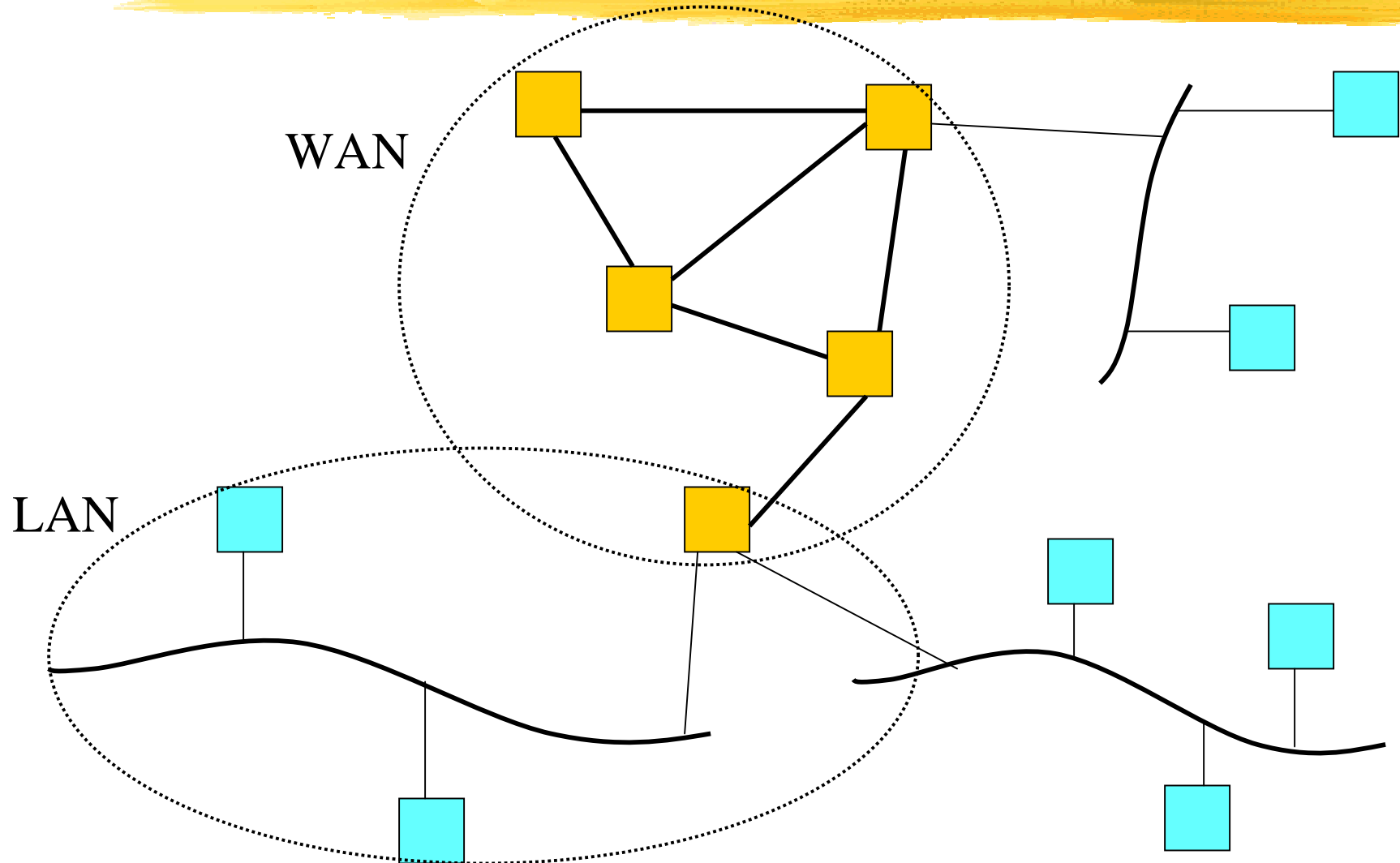
- ATM technology tries to achieve this.
- This course:
  - Understand Internet.
  - Learn ATM principles.

# A small Internet



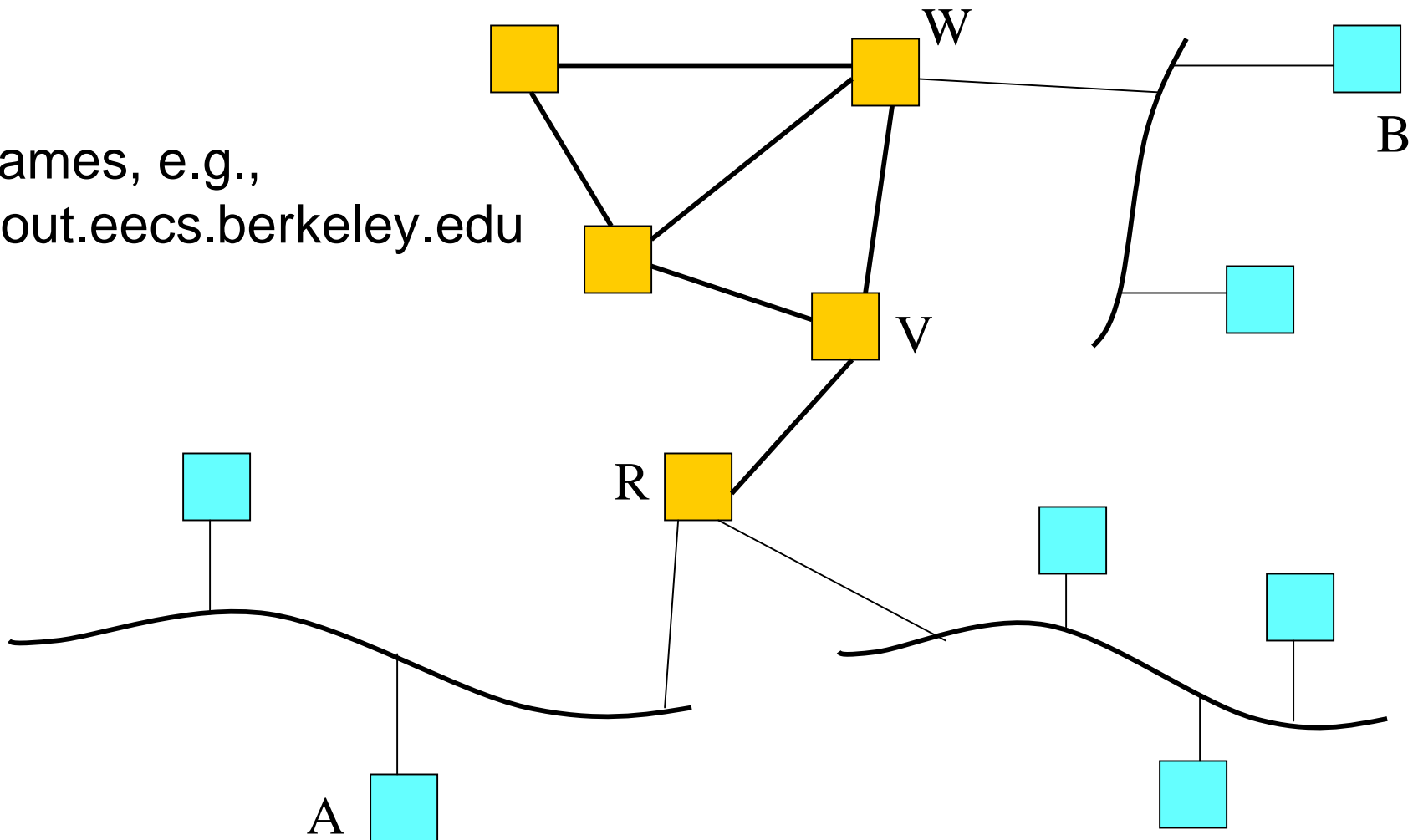


# A small Internet



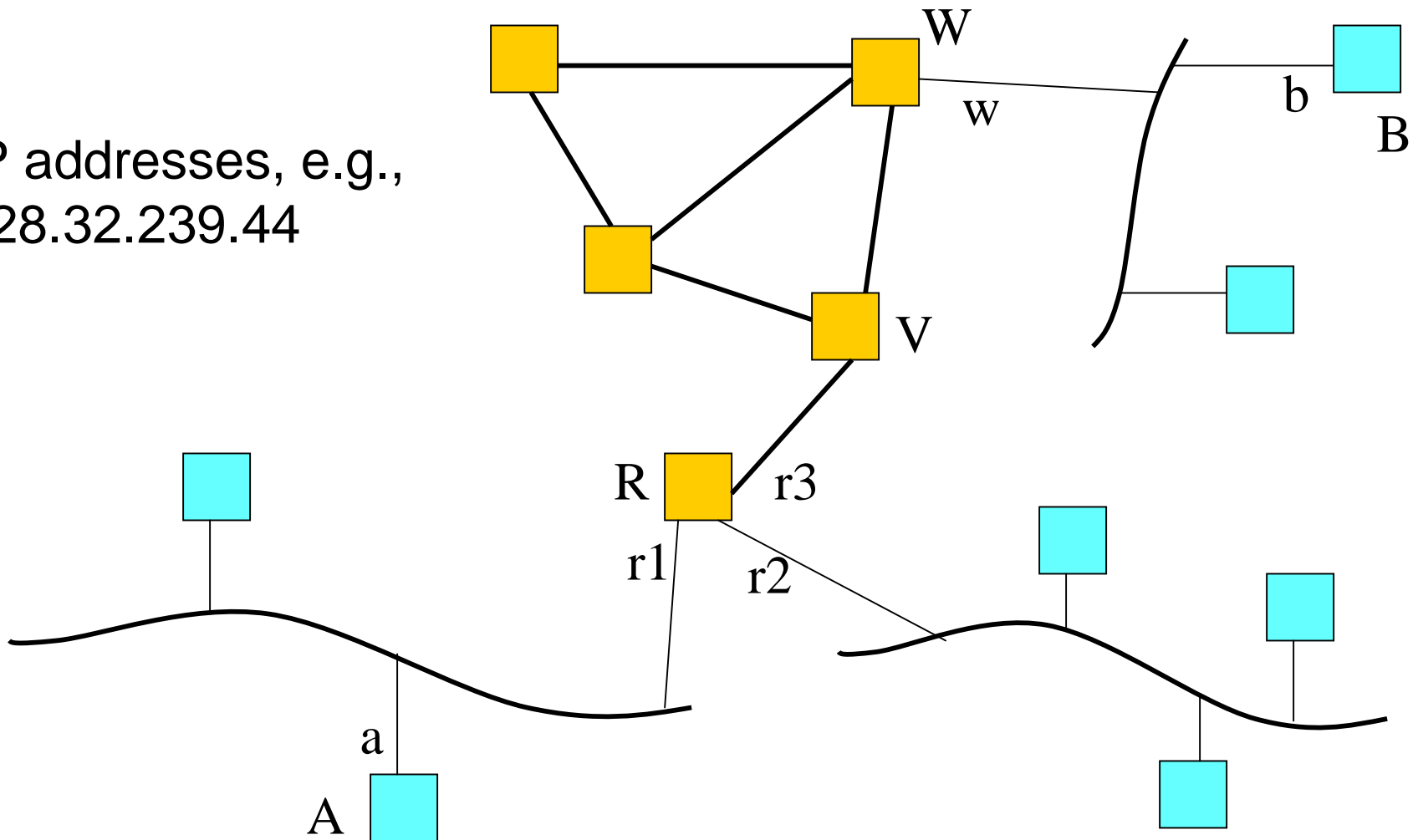
# A small Internet

Names, e.g.,  
stout.eecs.berkeley.edu

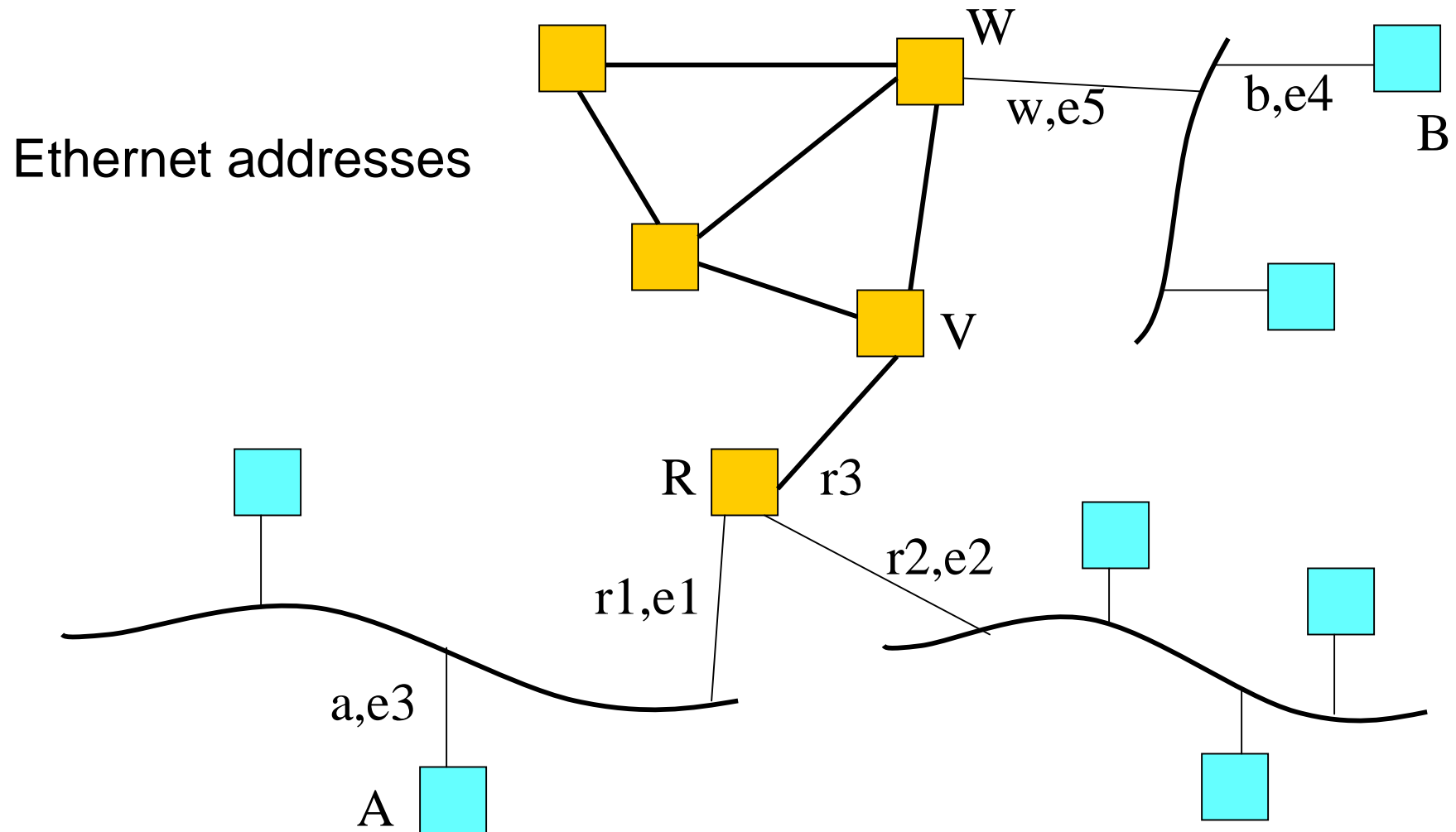


# A small Internet

IP addresses, e.g.,  
128.32.239.44

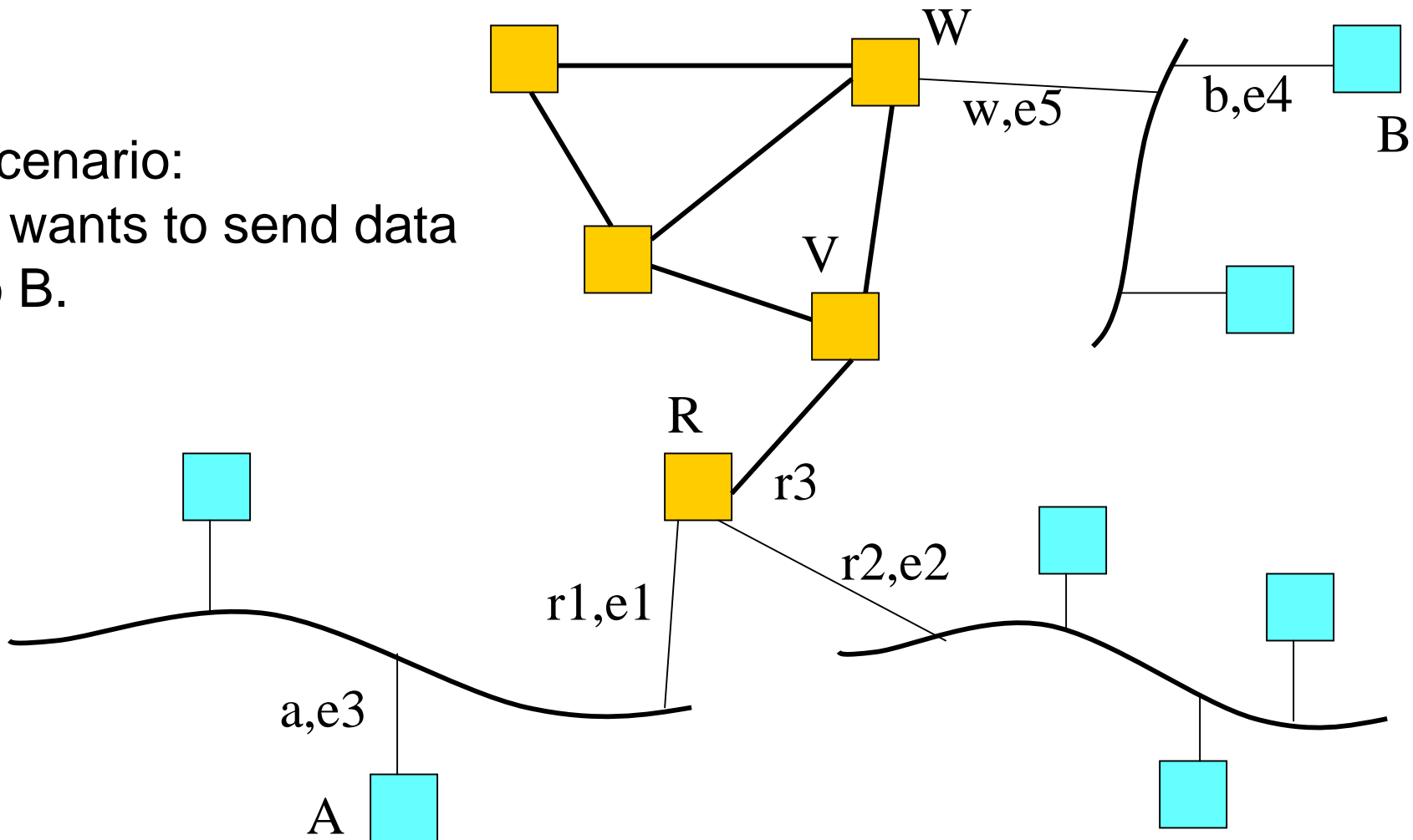


# A small Internet



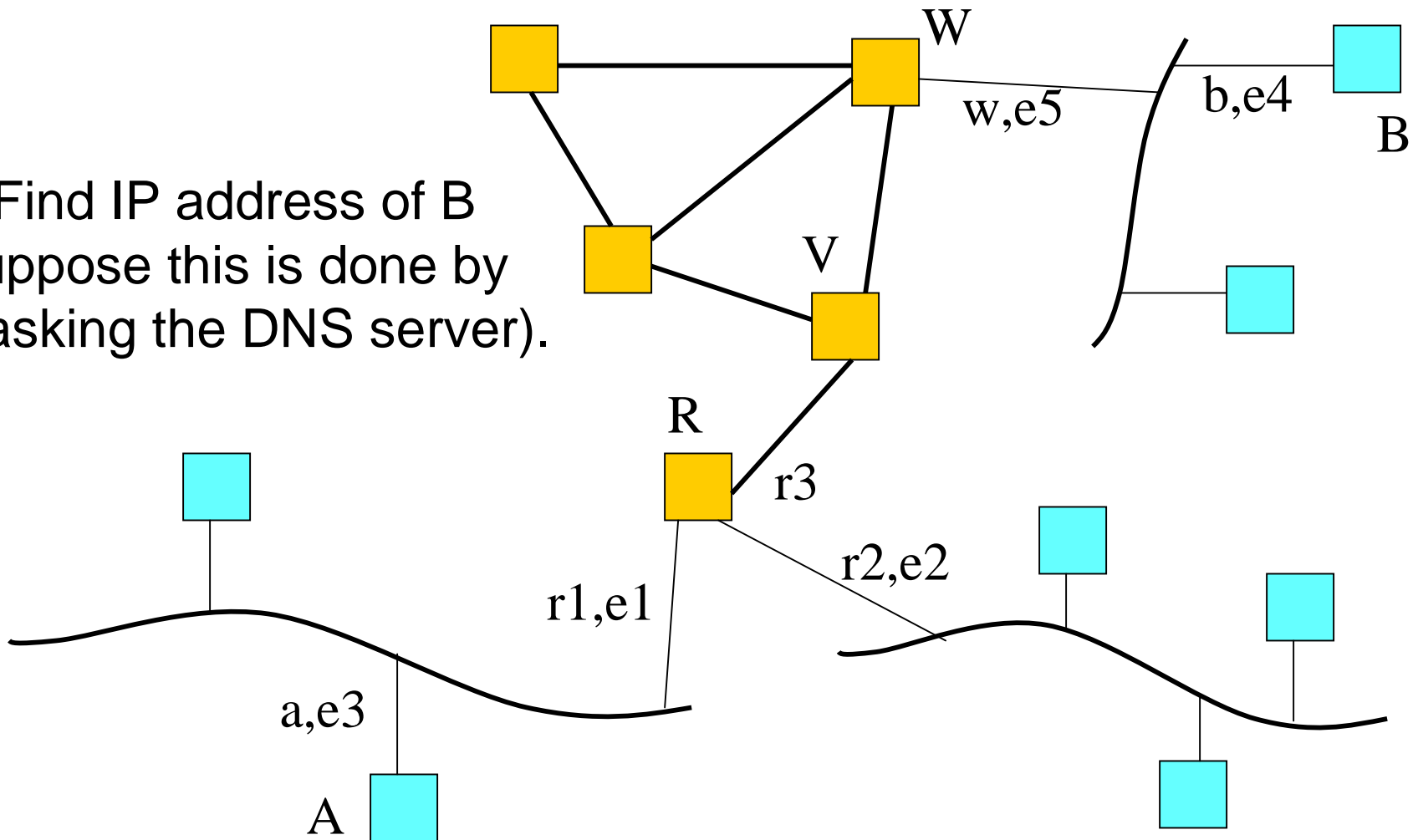
# A small Internet

Scenario:  
A wants to send data  
to B.



# A small Internet

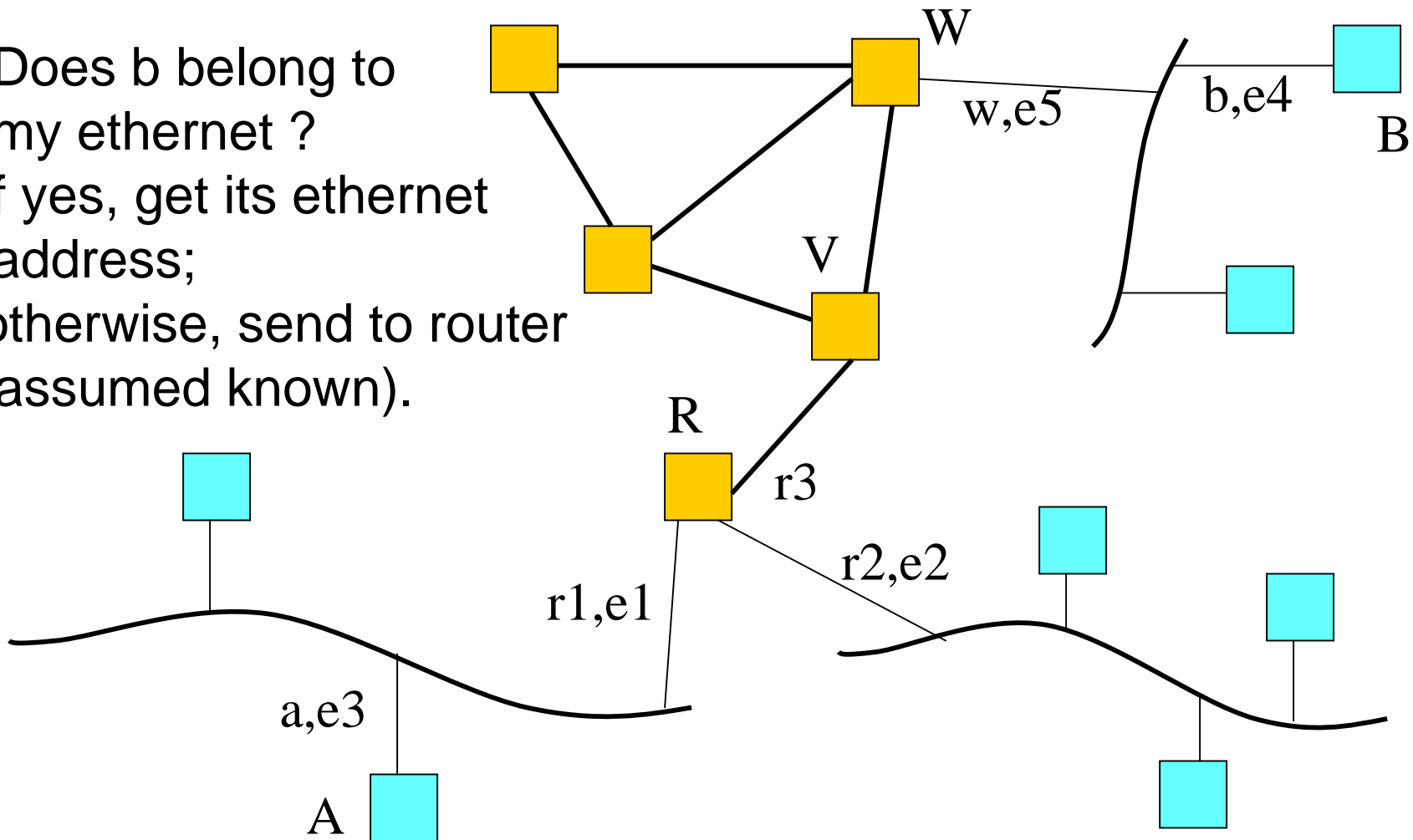
1. Find IP address of B  
(suppose this is done by  
A asking the DNS server).



# A small Internet

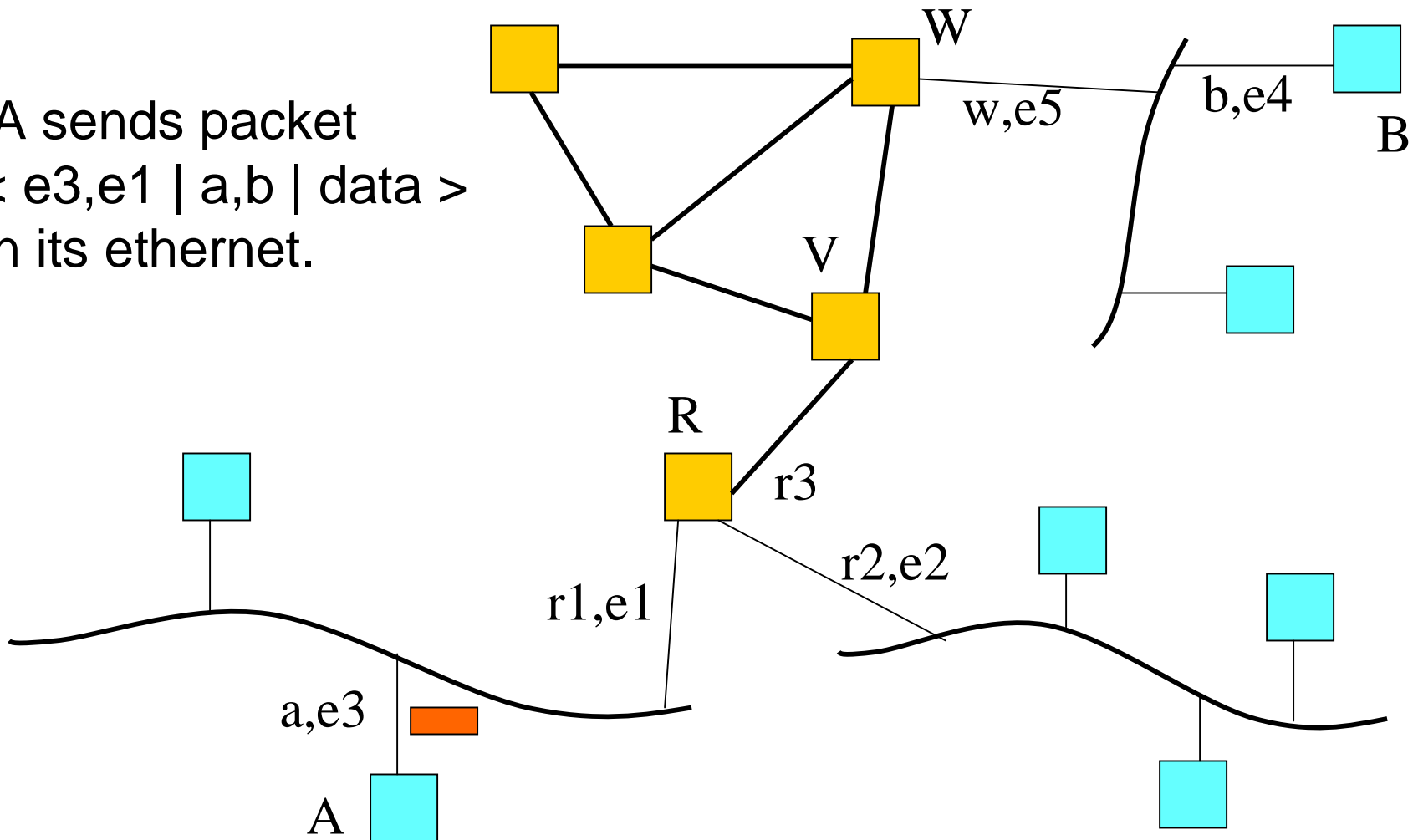
2. Does b belong to my ethernet ?

- if yes, get its ethernet address;
- otherwise, send to router (assumed known).



# A small Internet

3. A sends packet  
< e3,e1 | a,b | data >  
on its ethernet.

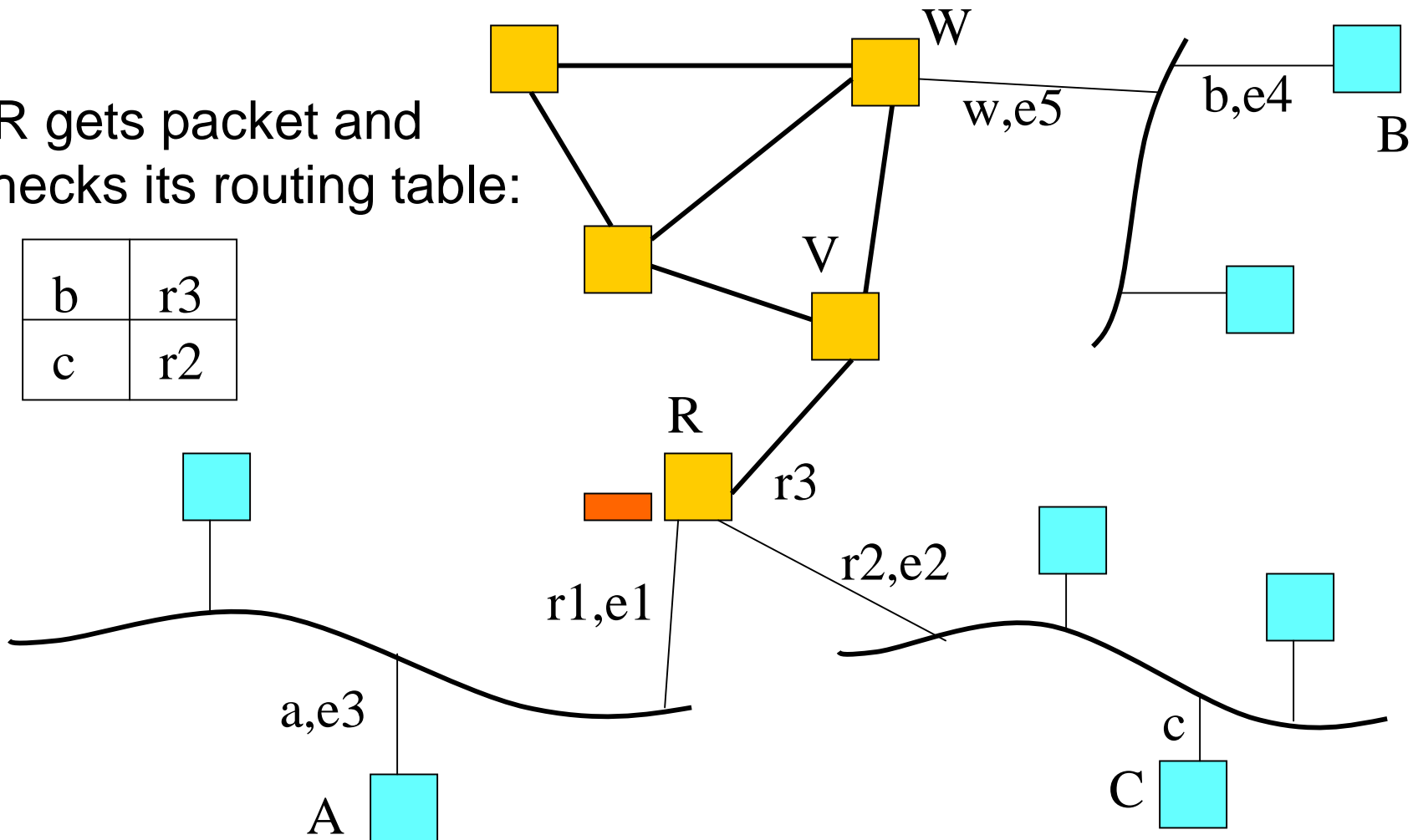




# A small Internet

4. R gets packet and checks its routing table:

b	r3
c	r2



# A small Internet

5-... R sends packet  
< ... | a,b | data >  
to router V.  
V checks its table, etc.

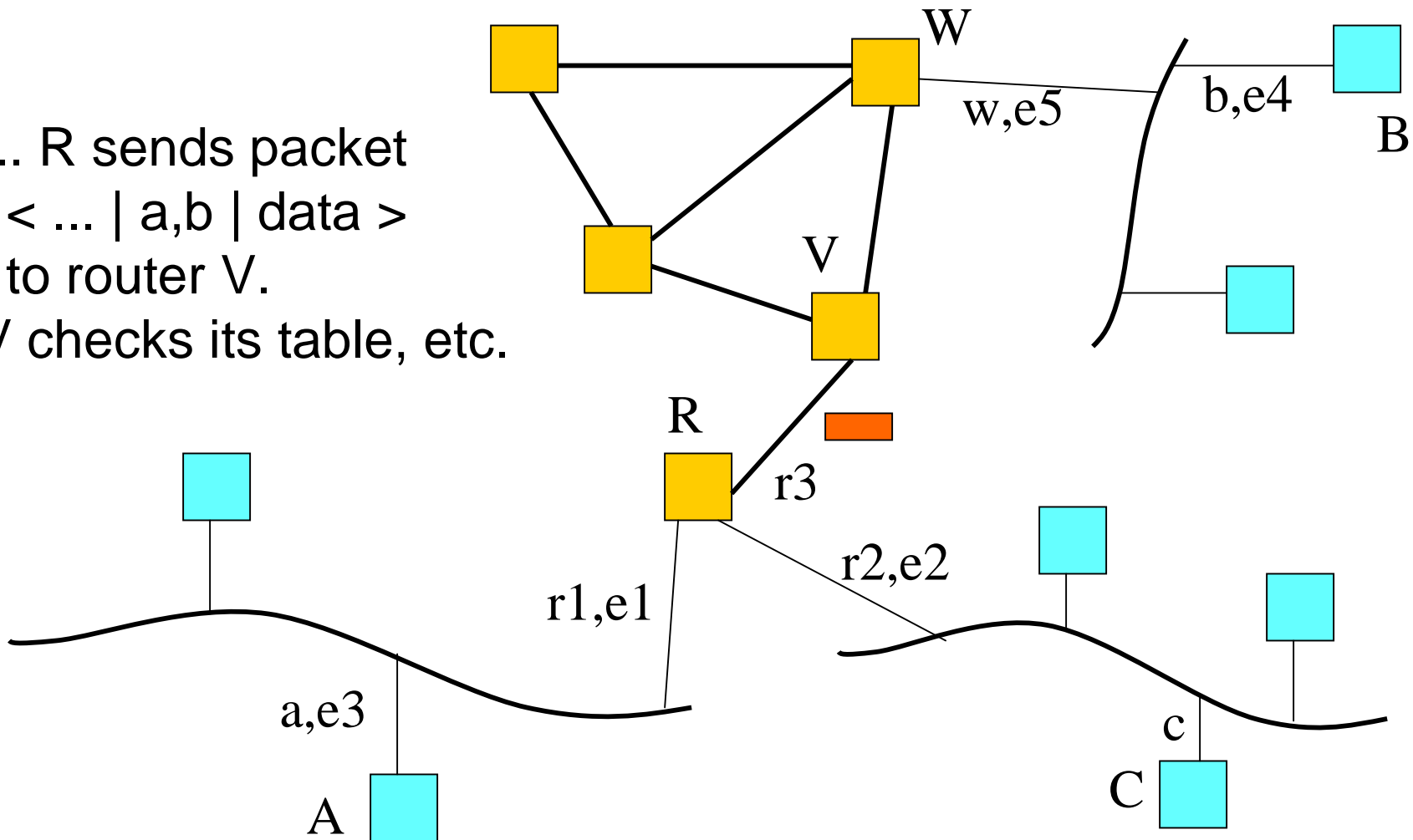
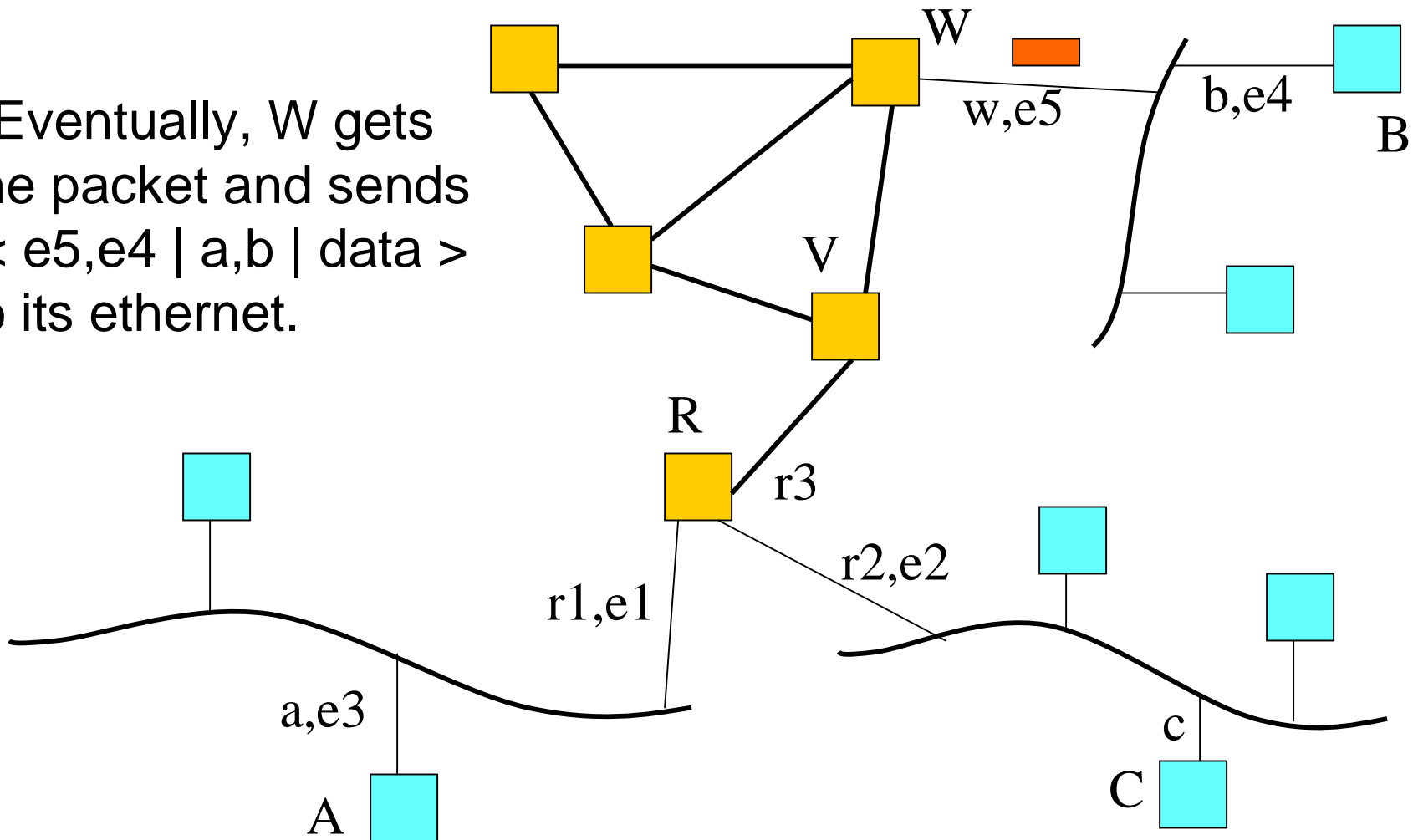


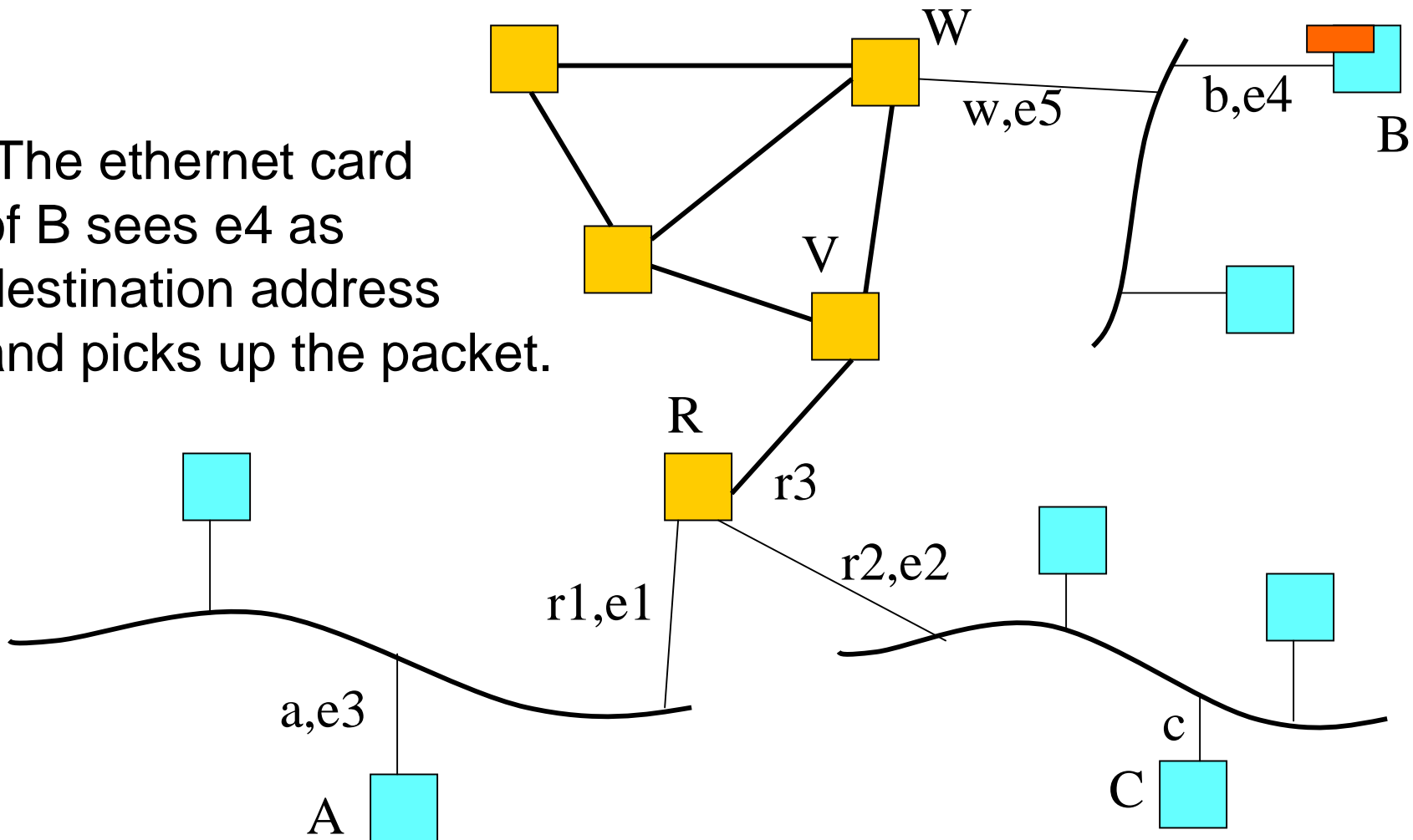
Figure 1. The effect of the number of trials on the mean number of correct responses for the 100% condition. The number of correct responses was significantly higher than the number of incorrect responses for all conditions.

6. Eventually, W gets the packet and sends  $\langle e5, e4 \mid a, b \mid \text{data} \rangle$  to its ethernet.



# A small Internet

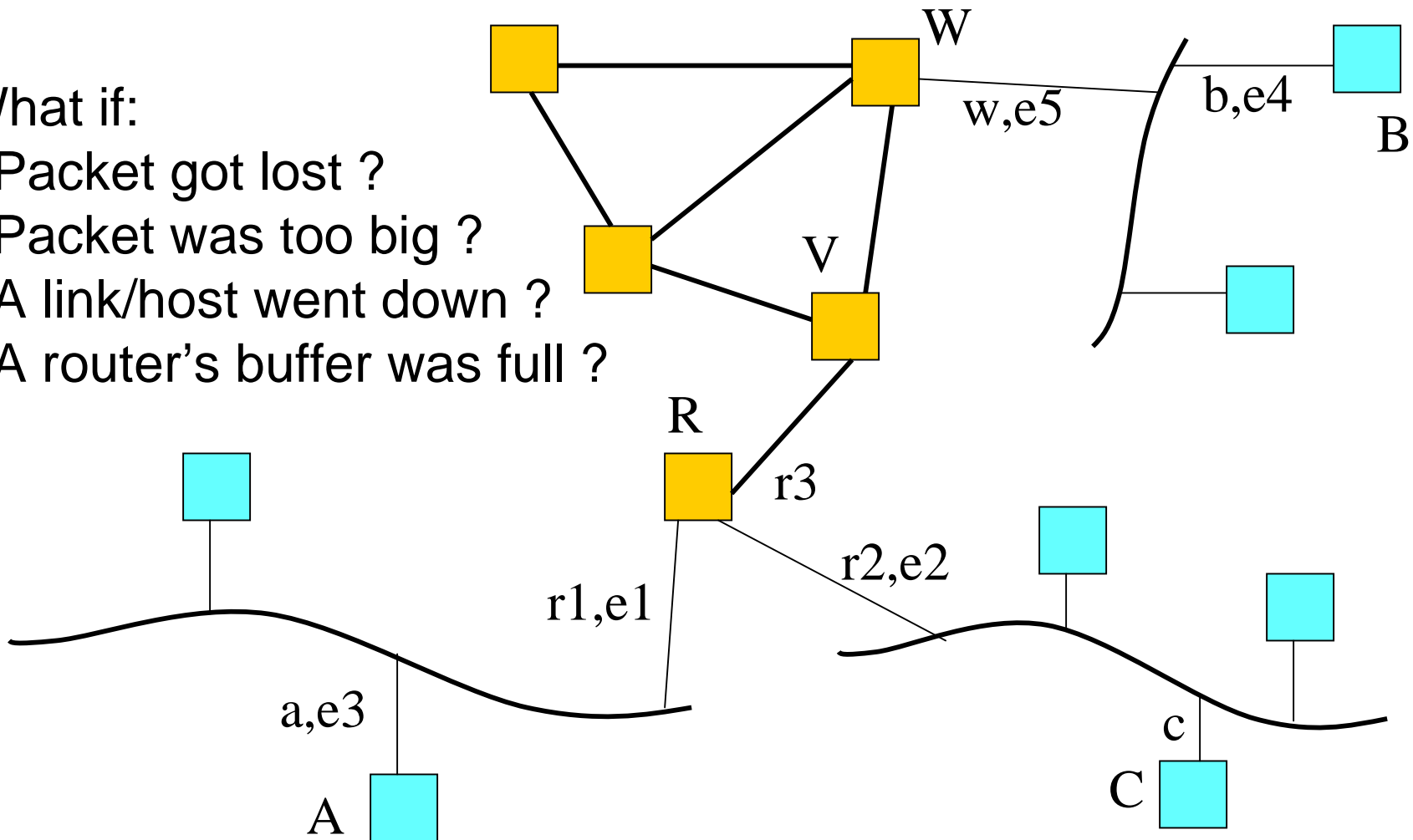
7. The ethernet card of B sees e4 as destination address and picks up the packet.



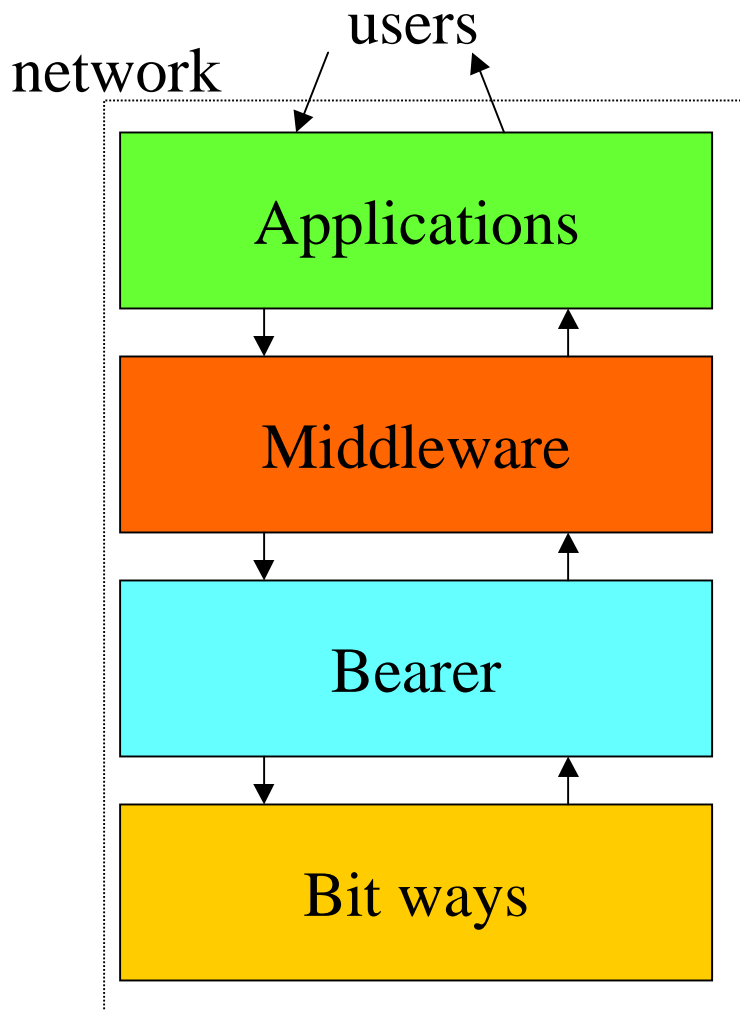
# A small Internet

What if:

- Packet got lost ?
- Packet was too big ?
- A link/host went down ?
- A router's buffer was full ?



# Layered Architecture



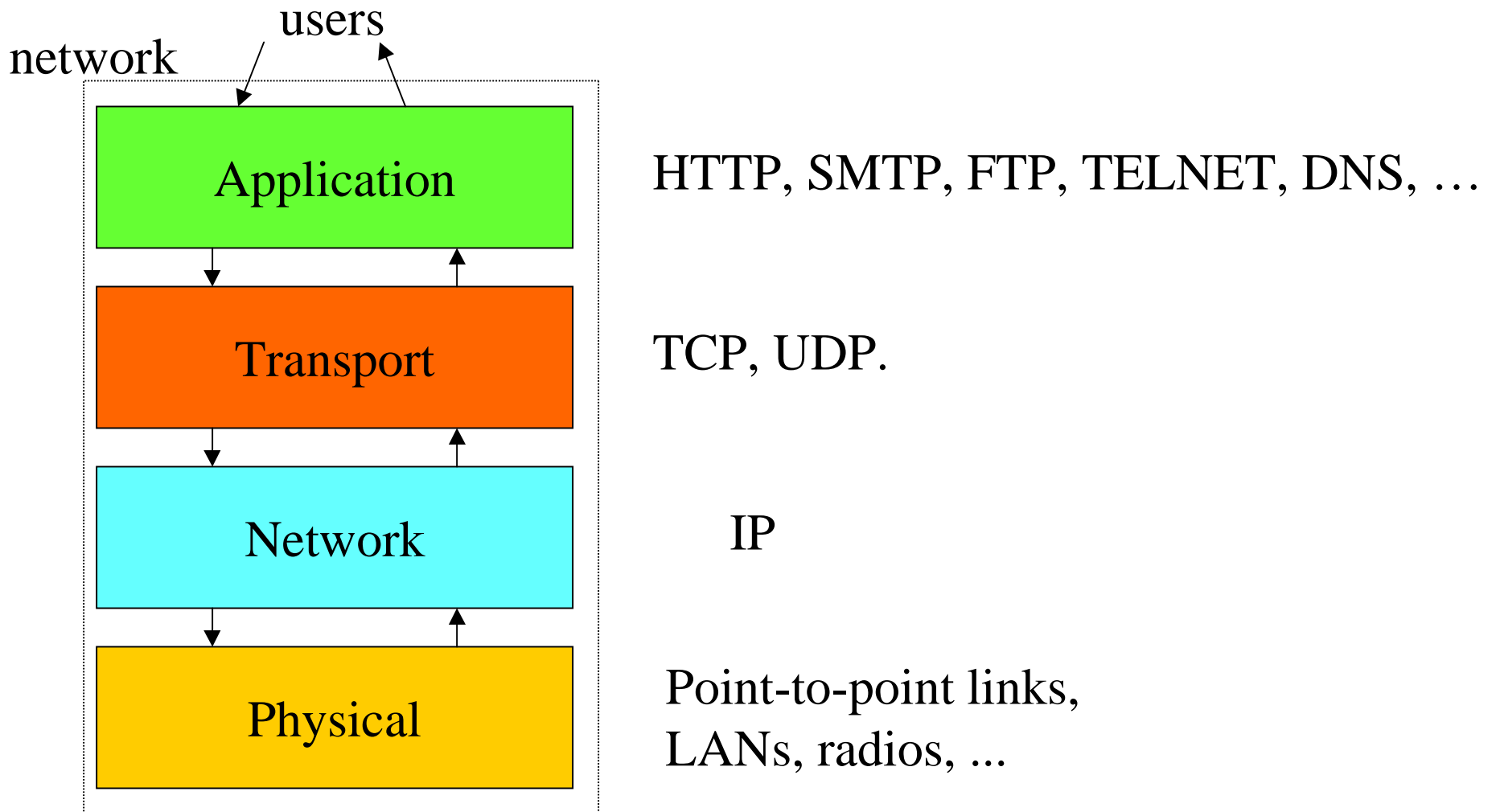
Web, e-mail, file transfer, ...

Reliable/ordered transmission, QOS, security, compression, ...

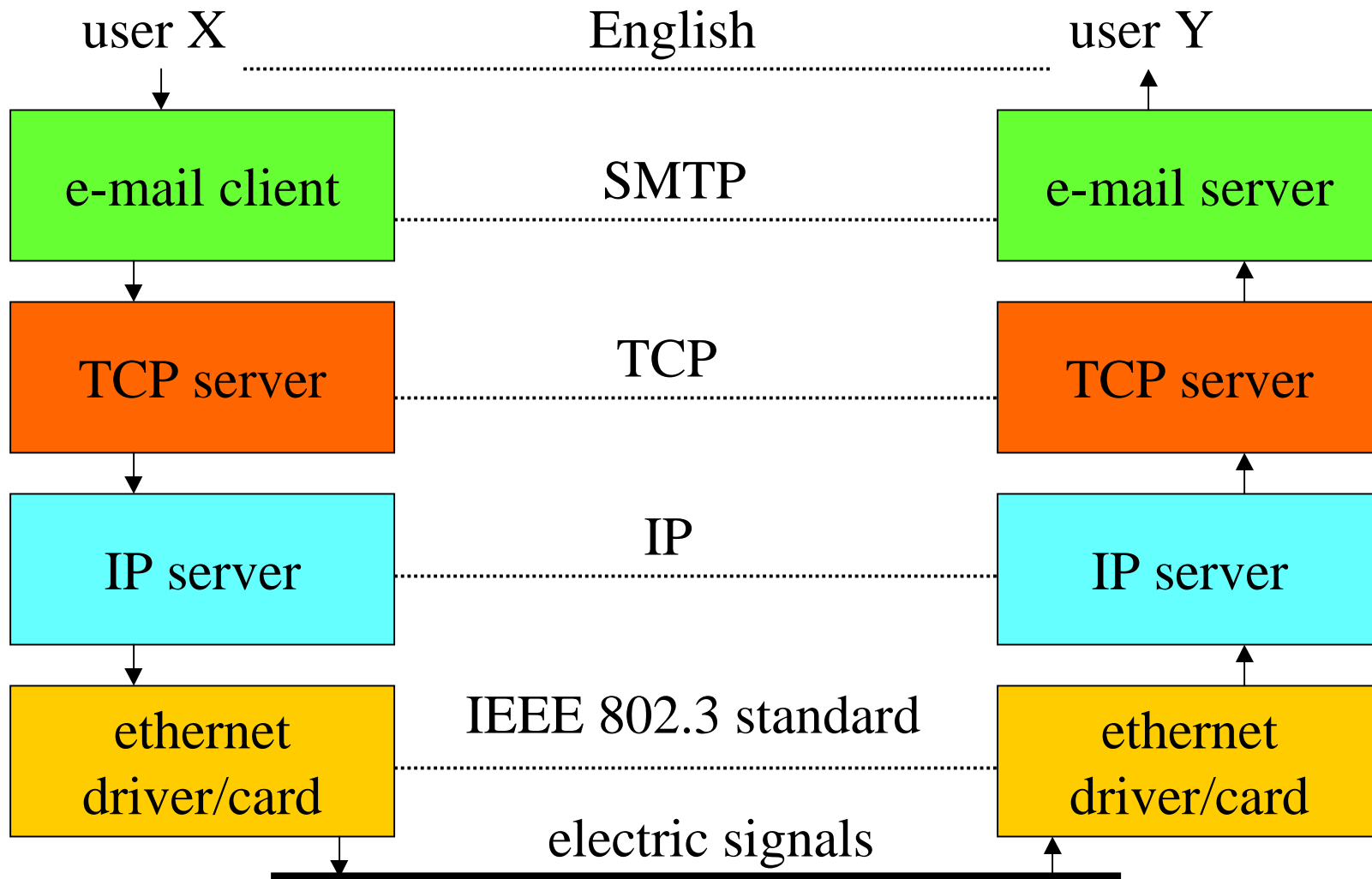
End-to-end transmission, resource allocation, routing, ...

Point-to-point links, LANs, radios, ...

# Internet protocol stack

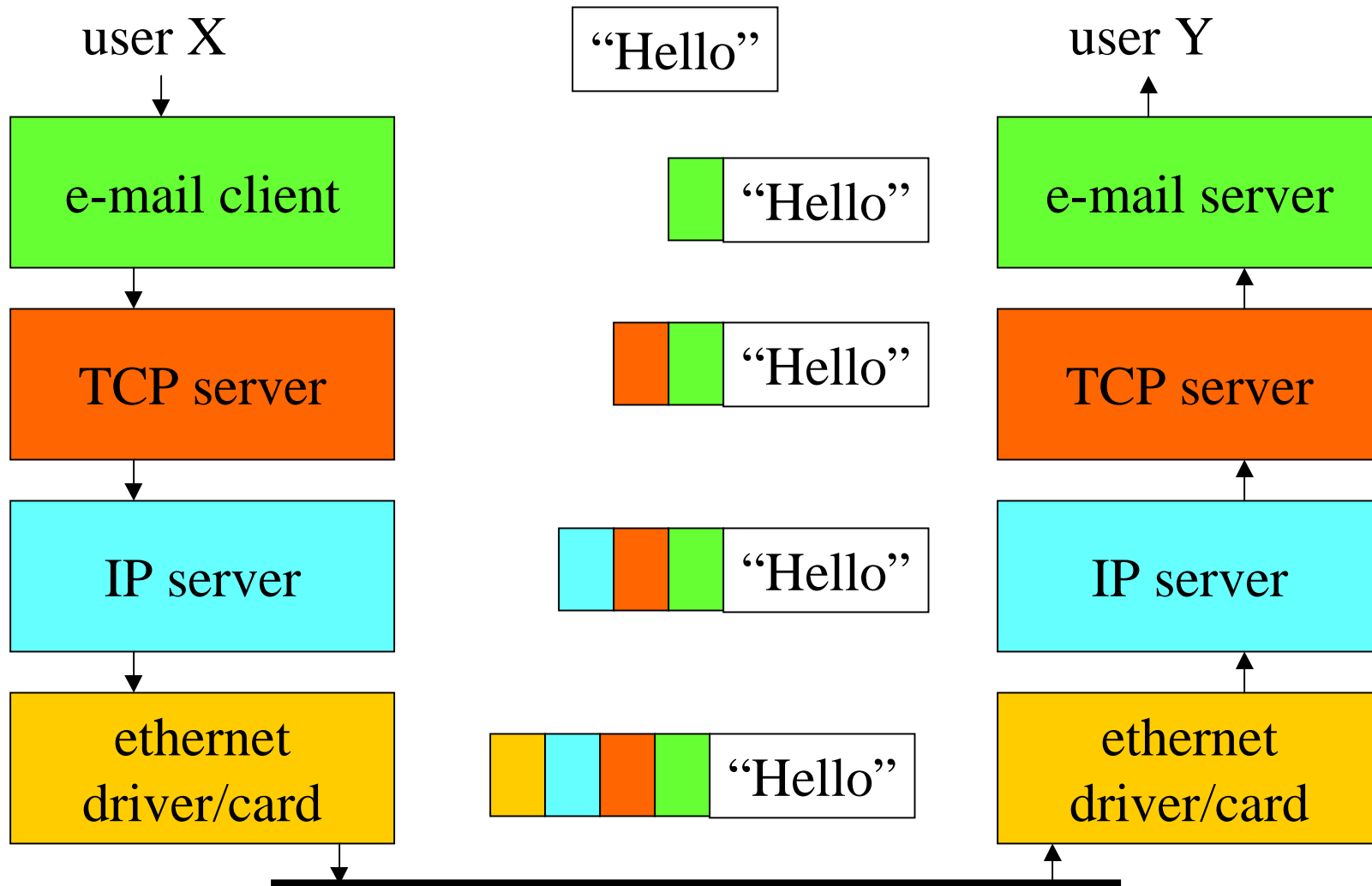


# Protocol stack

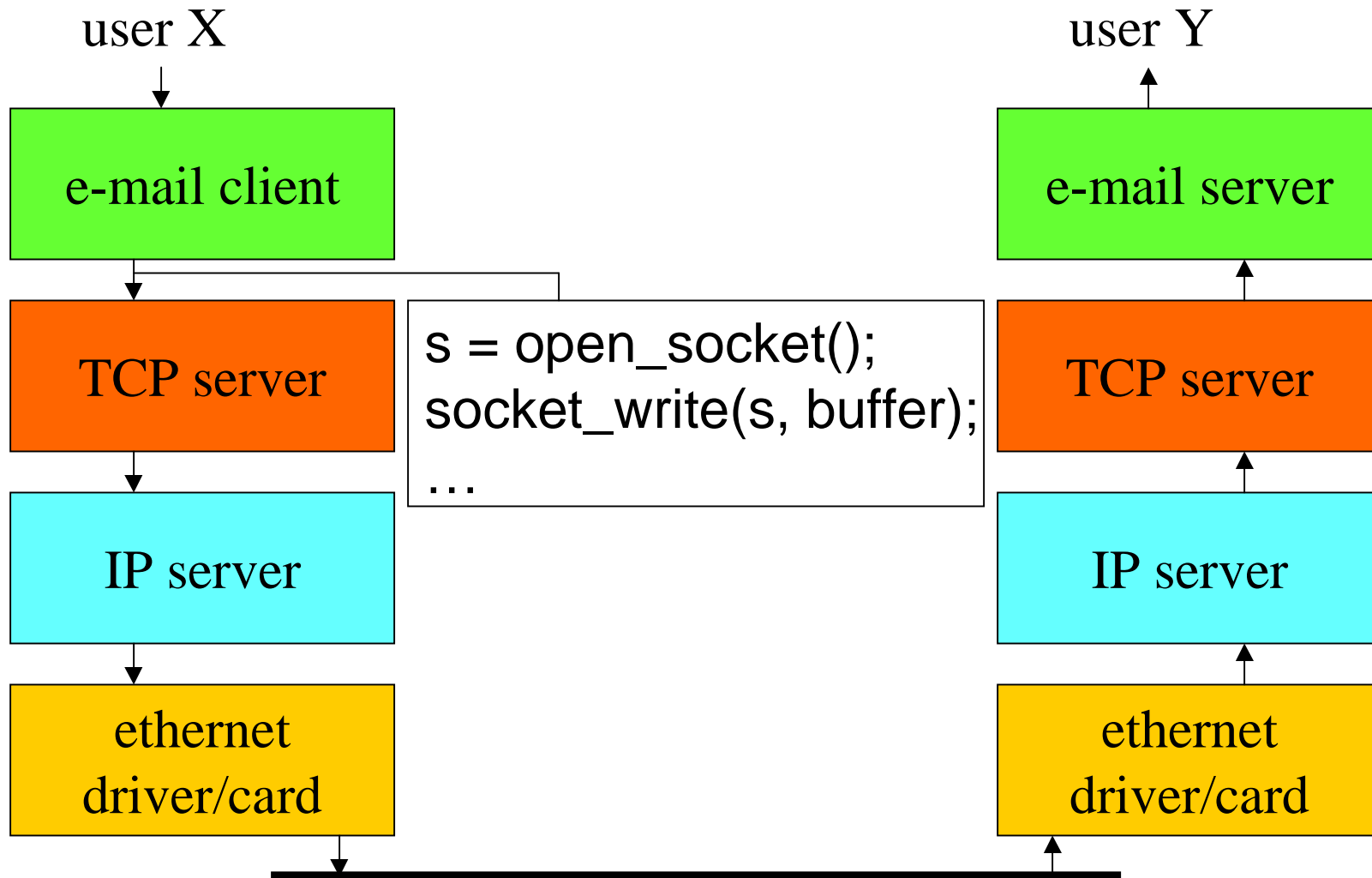




# Protocol encapsulation

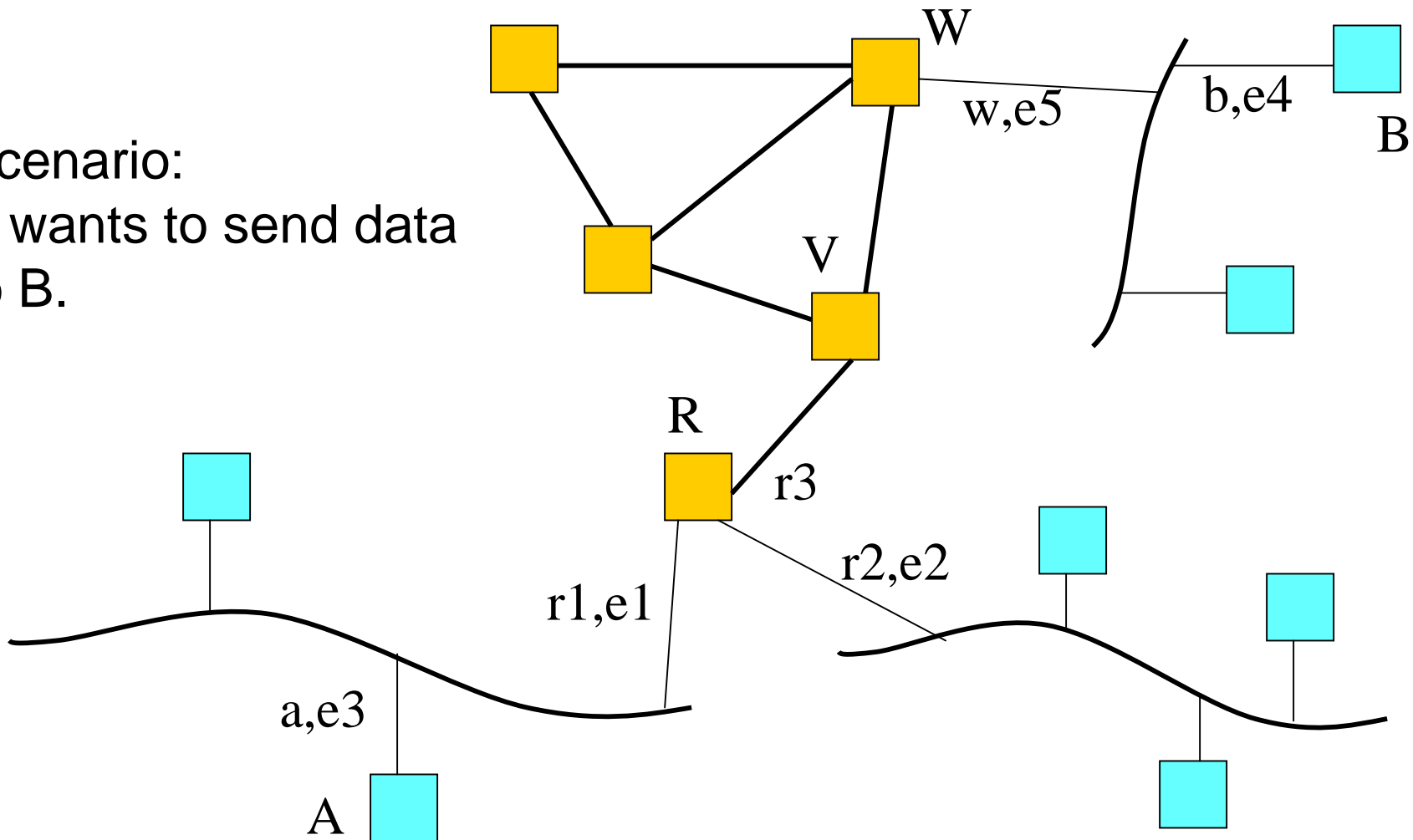


# Protocol interfaces

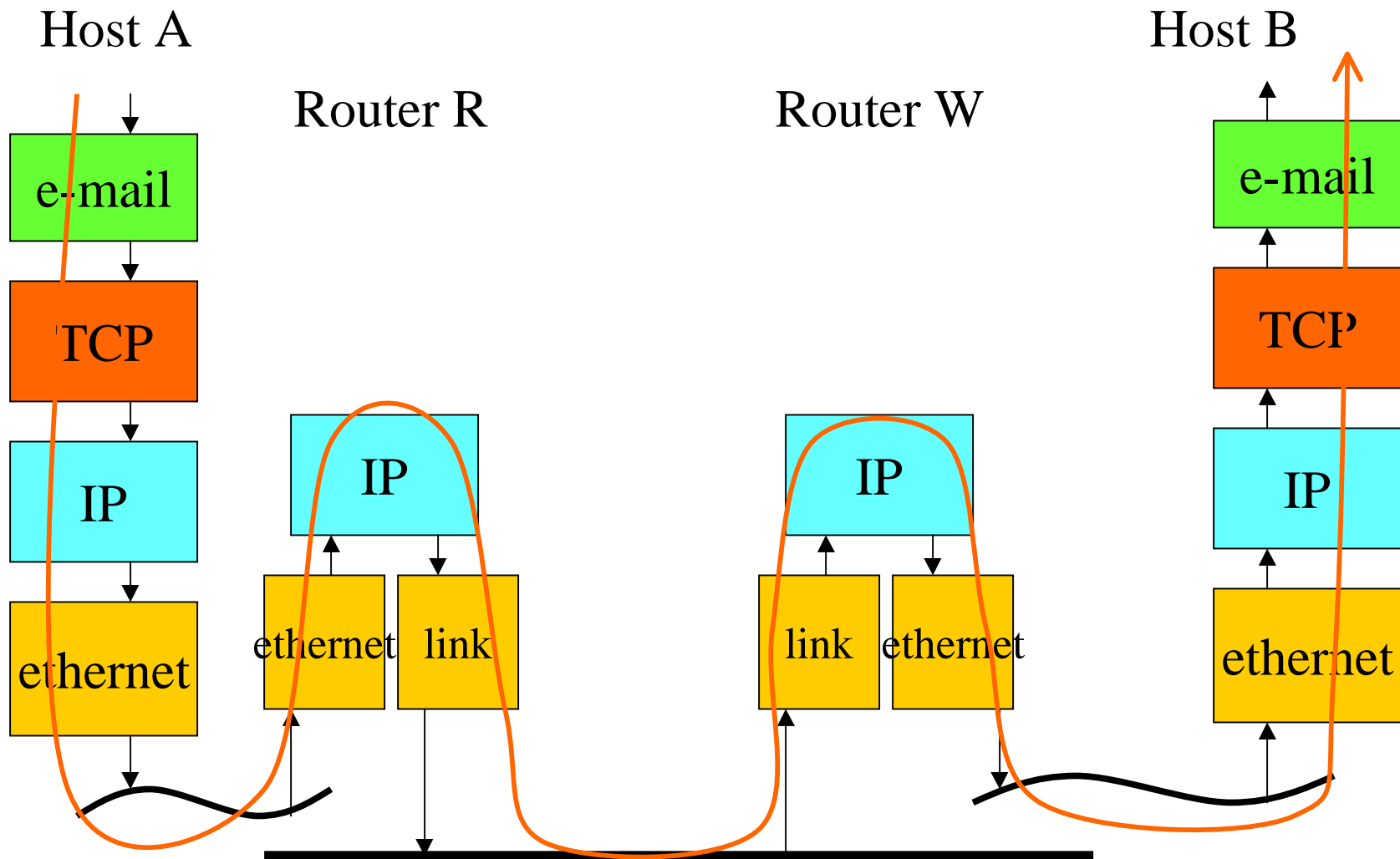


# A small Internet

Scenario:  
A wants to send data  
to B.



# Protocol stack: packet forwarding



# Layered Architectures



- Break-up design problem into smaller, more manageable problems.
- Modular design: easy to extend/modify.
- Difficult to implement (careful with interaction of layers for efficiency).

# Syllabus:



- Introduction (ends on Friday).
- Applications (e-mail, web, etc).
- Internet: architecture, protocols, addressing, routing.
- LANs (ethernet, token rings, wireless).
- ATM (quality of service).
- Reliable-transmission protocols (error correction, ordered transmission, etc).

# **Syllabus (continued):**



- Congestion control.
- Physical layer: copper, fiber, radio.
- Internet programming (sockets, etc) and network simulation (perhaps).
- Security.
- Compression.
- Special sessions: invited people will talk about interesting projects in communications going on in UCB.