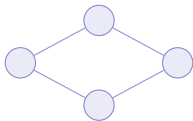


Computer Systems & Networks

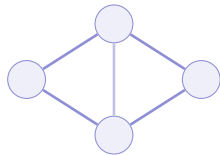
AP Computer Science Principles ■ Unit 4

AP Computer Science Principles



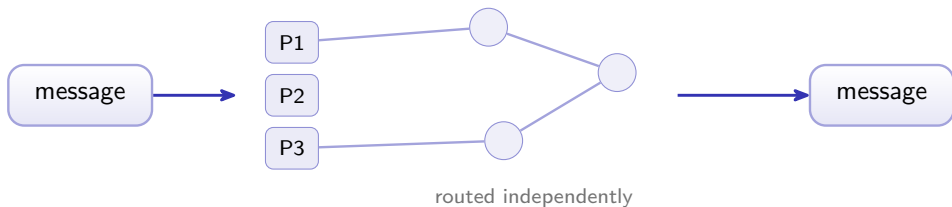
What we will cover

- 1 The Internet
- 2 Fault tolerance
- 3 Parallel & distributed computing





Data travels as packets



The **Internet** is a network of networks cooperating through shared **protocols**. Data splits into **packets**, each addressed by an **IP address** and reassembled in order. The **World Wide Web** is one service on top.



Packets carry data across networks

Protocols, and the two properties of the Internet's design

protocols

agreed rules for communication – without them two machines cannot interpret each other's bits

IP

Internet Protocol: **addresses and routes** packets toward a destination

TCP

reassembles packets **in order** and re-requests any that were lost

open

built on standards **anyone can use** – no permission needed to join or build on

scalable

it keeps working as billions of devices join; **bandwidth** is the bits per second a connection carries

Openness and scalability are **design decisions**, not accidents. Both are exam answers to “why has the Internet grown as it has?”



Fault tolerance

Fault tolerant

redundancy : extra paths/copies,
a backup takes over

Single point of failure

one part whose failure stops everything

The Internet is fault tolerant because many routes mean a broken link is bypassed. Redundancy costs money, so designers weigh it against reliability.



Routers forward packets toward their destination



Parallel & distributed computing

Sequential

one step at a time

Parallel

several **processors**, one machine

Distributed

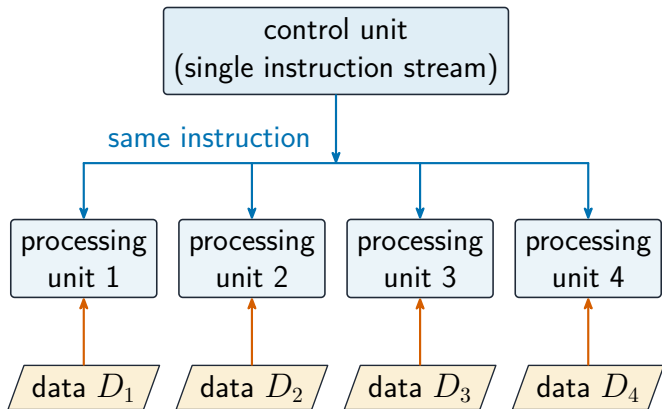
many separate machines

Speedup = $\frac{\text{one-processor time}}{\text{many-processor time}}$. The **sequential part** caps it – a 100 s task with 20 s sequential can never beat 5×



Distributed systems run across many machines

Parallel computing



Parallel computing: many processors work at the same time

Speedup, and the limit on it

sequential computing

one step after another, on one processor

parallel computing

several steps at the same time, on several processors

speedup

the **sequential** time divided by the **parallel** time

the limit

parts that **must** run in sequence cannot be sped up by adding processors, so speedup is bounded by that fraction

the consequence

a task that is 30% sequential can never exceed a speedup of about 3.3, however many processors you add

Doubling the processors never halves the time. Say *why* – the sequential portion – and the mark is yours.

Key points

The **Internet** routes independent **packets** via protocols

Redundancy makes a system **fault tolerant**

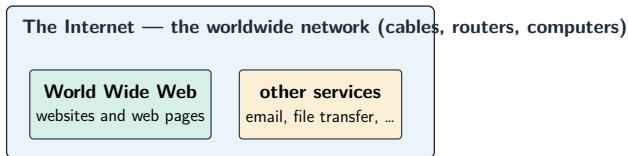
Parallel = one machine;
distributed = many

The **sequential part**
caps parallel speedup

The Internet is not the Web

The **Internet** is the *network of networks* – the wires, routers and protocols. The **World Wide Web** is *one service running on top* of it.

Email, video calls and games also ride the Internet *without* touching the Web.



one is the road, the other is one kind of traffic

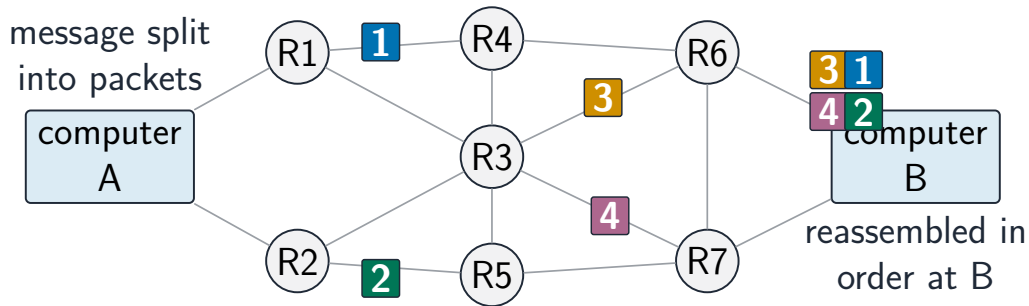
Worked example – the limit of parallelism

A 40 s sequential part + a 60 s parallelizable part

On one processor: $40 + 60 = 100$ s. Spread the parallel part over 3 processors: $\frac{60}{3} = 20$ s, so the total is $40 + 20 = 60$ s and the speedup is $\frac{100}{60} \approx 1.67$.

The 40 s sequential part is a **floor**: even with **infinitely many** processors the task can **never** finish in under 40 s.

Check yourself, in detail



Packet switching sends packets by independent routes, so the network tolerates a failed link

Exam tips

- Describe how data travels in **packets** over a redundant, fault-tolerant network of independent routers.
- Know that **protocols** (IP, TCP, HTTP) are agreed rules, and that open standards let different systems interoperate.
- Explain **scalability** and how the Internet grows without central control.
- Contrast **bandwidth** (rate) with **latency** (delay), and describe the **DNS** name-to-address lookup.
- Discuss the **digital divide** and security basics (encryption, authentication) in plain terms.



Unit 4 – key takeaways

1

Packets

redundant, fault-tolerant network
of routers

2

Protocols

IP, TCP, HTTP – open standards
interoperate

3

Scale

Internet grows without central
control

4

Terms

bandwidth vs latency; DNS name-
to-address

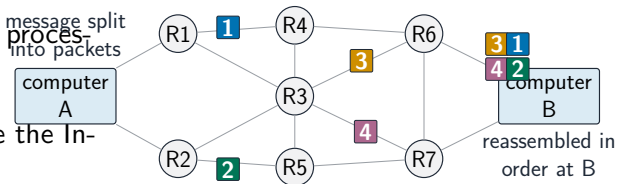
Check yourself

1 Is the Web the same thing as the Internet?

2 A 40 s sequential part: can 1000 processors beat 40 s?

3 Why does packet switching make the Internet fault tolerant?

4 What is redundancy in a network?



Answers 1. no – the Web is one service on the Internet 2. no – it is a floor 3. packets reroute around a failure 4. more than one path between two points