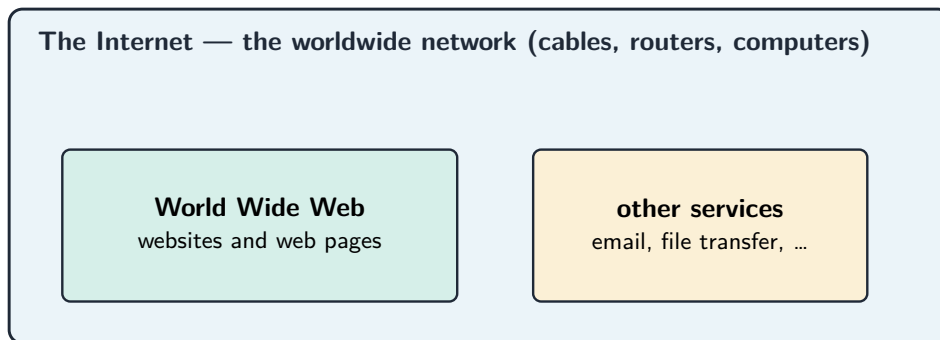


Computer Systems and Networks

AP Computer Science Principles

The Internet

The **Internet** 互联网 is a global network of networks. Data travels as **packets** 数据包—small chunks that are sent separately and reassembled at the destination. Two ideas make it work at scale:



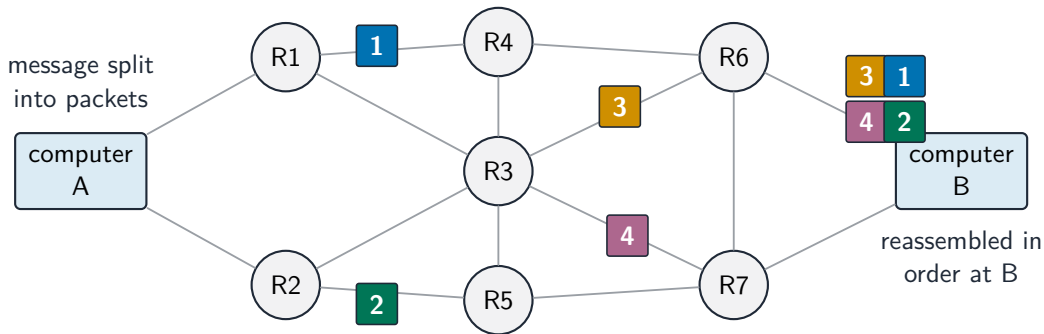
The internet is the worldwide network; the web is one service running on it

- **Protocols** 协议 are agreed rules for communication. **IP** (Internet Protocol) addresses and routes packets; **TCP** reassembles them in order and re-requests lost ones; **HTTP** carries web pages; **DNS** translates a name like `example.com` into an IP address.
- **Redundancy** 冗余 and routing: there are **many** possible paths between two points, so if one path fails, packets take another. This makes the Internet **fault-tolerant** 容错.

The Internet is designed to be **open** and **scalable**—built on standards anyone can use, so it keeps working as billions of devices join. **Bandwidth** 带宽 is the amount of data a connection can carry per second.

Fault Tolerance

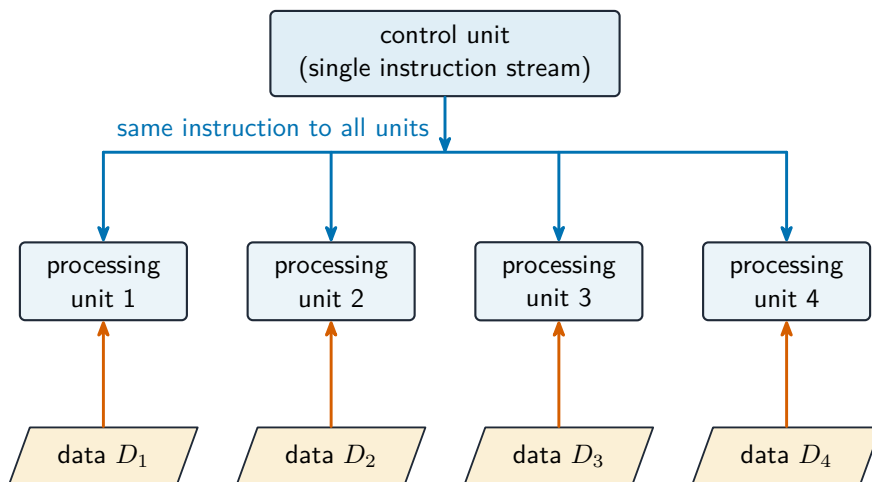
A system is **fault-tolerant** if it keeps working even when part of it fails. The Internet achieves this through **redundant** connections: because packets can be routed along multiple paths, the failure of one router or cable does not stop communication—traffic simply reroutes. Fault tolerance costs extra resources (the redundant paths) but greatly improves **reliability** 可靠性. A single path with no backup is not fault-tolerant.



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

Parallel and Distributed Computing

- **Sequential computing** 顺序计算 runs one operation at a time.
- **Parallel computing** 并行计算 splits a task into parts that run **at the same time** on multiple processors, finishing faster.
- **Distributed computing** 分布式计算 uses **many computers** connected by a network to work on one problem –essential for problems too big for a single machine.



Parallel computing: many processors work at the same time

A parallel solution's **speedup** 加速比 is the sequential time divided by the parallel time. Speedup is limited: parts that **must** run in sequence cannot be sped up by adding processors, so doubling the processors rarely doubles the speed.

Worked example. A task has a part that **must** run sequentially, taking 40 seconds, plus a parallelizable part that takes 60 seconds on one processor –so on a single processor the whole task takes $40 + 60 = 100$ seconds. Spread the parallel part across 3 processors and it takes $\frac{60}{3} = 20$ seconds, so the total parallel time is $40 + 20 = 60$ seconds and the speedup is $\frac{100}{60} \approx 1.67$. The 40-second sequential part is a floor: even with infinitely many processors the task can never finish in under 40 seconds.

Exam skill: given the times for the sequential and parallel portions of a task, be able to calculate the total parallel time and the speedup.

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.