Engineer

Hardware

the physical parts of a computer

What it is

Hardware is the part of a computing system made of matter rather than of code: the processor that carries out instructions, the memory that holds the work in progress, the disks or flash that keep data while the power is off, the network interface, the power supply and the board that wires them together. Screens, keyboards, cables, switches and routers count as well. A physical server with no virtualisation layer between it and its operating system is often called *bare metal*.

NIST's glossary defines hardware as the material physical components of a system and sends the reader on to software and firmware, because the three are defined against each other. Software, in the same glossary, is programs and data that may be written or modified while they run. Firmware is programs and data held in hardware, usually in read-only or programmable read-only memory, so that they cannot be written or modified while they run. Firmware is what starts a machine before any operating system is loaded, and smaller pieces of it run inside disks, network cards and battery controllers. NIST's guidelines on platform firmware treat the hardware and firmware needed to boot and operate a system as one thing, which they call the platform.

Each part has a capacity that software can use well or badly but cannot raise. The processor does a fixed amount of work per second and is shared by everything running. Memory is fast, limited and emptied when the power goes. Storage is slower and larger, keeps its contents through a power cut, and wears with use. The network interface moves a fixed number of bits per second. A system is as fast as whichever of these runs out first for the work it is given.

Most servers today run on hardware nobody at the company has seen. A virtual machine rented from a cloud provider is a share of a physical machine in the provider's data centre: a hypervisor divides that machine's processors, memory

and disks between the virtual machines placed on it and keeps each to its share. The customer chooses sizes from a price list, and the provider does the cabling, the spare parts and the replacing.

Use the concept in a decision

Put “What is hardware?” to work in a real decision. We can help relate the concept to the system you are building or buying. The laptop, the printer, the login and the mailbox, sorted quickly and in plain language, without anyone feeling foolish.

Explore our services Get in touch →

Test the assumptions behind the term

Use the explanation and linked work to frame questions for your project. We can examine the assumptions with you before a tool is chosen. Established a Technical Support department — 10,000 clients served by a support department built from scratch

Established a Technical Support department Get in touch →

Why it matters

Hardware sets the ceiling. Software decides how much of a machine’s capacity turns into useful work, but no setting gives a machine more memory than it has. Knowing which part is full, the processor, the memory, the disk or the network, tells you what to change, and often that the answer is software.

Hardware is the slowest and most expensive thing to change. A program can be reinstalled in minutes. A part has to be chosen, ordered, fitted and paid for, and a larger rented machine costs more every month for as long as it runs. That makes it the last resort for performance, though it is often the first thing proposed.

Hardware is where things physically wear out. Fans, batteries, hinges, connectors and storage all have working lives. Everything above them can be rebuilt from a backup and a written description; the parts themselves have to be replaced.

And hardware holds data after software has let go of it. Deleting a file usually removes the file system’s record of where the content is, not the content itself. A disk that leaves the building, sold, returned under warranty or thrown out, can carry much of what was ever written to it.

What it looks like in practice

The portfolio’s hardware work starts at a repair bench. The account of diagnosing and repairing laptop hardware covers screens, hinges, keyboards, trackpads, motherboards and power, along with software on Linux, Windows and Mac. It

puts the skill in the diagnosis: a laptop that will not power on could be the charger, the board or the battery, and the repair is only as good as the guess about what is actually broken. So the care went into finding the real fault before touching anything.

This site runs on hardware at the other extreme, a machine it has never touched. The credits page describes the one small machine that does all of it: a single-core DigitalOcean server with 512 MB of memory and a 10 GB disk, standing in Toronto, Canada. The same page says Ansible describes that server as code, so it can be rebuilt rather than remembered. Off-host backup automation using restic is implemented separately; that implementation alone is not evidence that backups are active. When the hardware belongs to someone else, those two lines are the repair kit.

And the note on optimising a database before sharding it is largely about when not to reach for hardware. Spreading data across more servers spreads slow statements over more hardware without making one of them cheaper. Rows queueing for a lock are waiting for each other, not for hardware, and a bigger machine does not shorten the wait for one row. When many times more sessions are active than there are cores, a connection pool fixes that before any hardware does. The note's test for splitting is a machine full of good work, not one full of waiting.

Where it goes wrong

Hardware is bought for a software problem. A slow system looks like a small one. Before paying for a larger machine, find out which part is full. On Linux, three commands give a first answer: how many processors the system can use, how much memory is in use and available, and five one-second samples of where processor time goes.

```
nproc
free -h
vmstat 1 5
```

In the `vmstat` output, `us` and `sy` are time spent in programs and in the kernel, `id` is idle and `wa` is time waiting for storage. A processor that is mostly idle while requests wait means the wait is somewhere else: a lock, a slow query, a

remote call or too many connections at once. More hardware leaves all of those where they were.

The wrong part is replaced. A symptom names a place, not a cause. A machine that will not start can be the power, the board or the battery; a slow disk can be a full one, a failing one or a busy one. Swapping parts until the fault goes away costs money and hides which part was at fault, so it returns. Test the cheapest explanation first, change one thing at a time, and confirm the fix by repeating whatever showed the fault.

A virtual processor is taken for a physical core. Sizing a rented machine by its count of virtual CPUs assumes each one is a core. DigitalOcean's documentation, for one, defines a vCPU as one hyper-thread of a processor core. On its shared-CPU plans that hyper-thread may be shared with other virtual machines on the same physical host, and when they are busy a machine can receive fractions of one, while memory, disk and network bandwidth stay dedicated. Measure under the load you expect at peak, and move to dedicated processors when timings vary more than the work can tolerate.

Data leaves with the disk. Deleting files or formatting a disk can leave its contents readable to anyone with ordinary recovery tools. NIST's guidelines for media sanitisation, revised in September 2025, define sanitisation as a process that "renders access to target data on the media infeasible for a given level of effort", and choose the technique by the sensitivity of the information. Decide how each disk will be sanitised before it is bought, and keep a record when it goes.

Firmware is left out of the updates. Operating system updates usually leave the firmware in the board, the disks and the network card as it was. NIST's platform firmware guidelines warn that a successful attack on that firmware could leave a system inoperable, perhaps permanently, and build their advice on protecting it from unauthorised changes, detecting the changes that happen and recovering quickly. Apply vendor firmware updates on a schedule, the way the rest of the system is patched.

Further reading

- hardware — NIST CSRC Glossary — the short definition, with its pointers to software and firmware.

- DigitalOcean: Choosing the Right CPU Droplet Plan — what a virtual CPU is at the provider this site runs on, and when shared processors are enough.
- NIST SP 800-88 Rev. 2, Guidelines for Media Sanitization — how to put the data on a disk out of reach before the disk leaves.
- NIST SP 800-193, Platform Firmware Resiliency Guidelines — protecting, detecting and recovering the firmware a machine boots from.

Glossary

Choose for your actual constraints

Describe the device or service, the observed symptoms and the data that must be preserved. We can establish the safest useful diagnostic step.

See what it costs Get in touch →

Make the terminology practical

Ask how “What is hardware?” applies to your organisation. A concrete example from your current system is enough to start the discussion.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/glossary/hardware/