1_9

Chapter 1: A Tour of Computer System

1.9 Important Themes

- A system is more than just hardware. It is a collection of intertwined hardware and systems software that must cooperate in order to achieve the ultimate goal of running application programs.

1.9.1 Amdahl's Law

- Amdahl's law - The main idea is that when we speed up one part of a system, the effect on the overall system performance depends on both how significant this part was and how much it sped up.
- $T_{new} = (1-\alpha)T_{old} + (\alpha T_{old})/k$
  - $$T_{old}$$
    - the time that a system in which executing some application requires.
  - $$\alpha$$
    - before improving, some part of the system requires a fraction α of
    $$T_{old}$$
    .
  - k - we improve the part of the system performance by a factor of k.
  - $$T_{new}$$
    - the new overall execution time.
- The speedup

$$S = T_{old}/T_{new} = \frac{1}{(1-\alpha)+\alpha/k}$$

Practice Problem 1.1

Suppose you work as a truck driver, and you have been hired to carry a load of potatoes from Boise, Idaho, to Minneapolis, Minnesota, a total distance of 2,500 kilometers. You estimate you can average 100 km/hr driving within the speed limits, requiring a total of 25 hours for the trip.

A. You hear on the news that Montana has just abolished its speed limit, which constitutes 1,500 km of the trip. Your truck can travel at 150 km/hr. What will be your speedup for the trip?

Solution:

$$\alpha = 1500/2500 = 0.6$$

$$k = (1500/100)/(1500/150) = 150/100 = 1.5$$

$$s = \frac{1}{(1-0.6)+0.6/1.5} = 1.25$$

B. You can buy a new turbocharger for your truck at [www.fasttrucks.com.](www.fasttrucks.com) They stock a variety of models, but the faster you want to go, the more it will cost. How fast must you travel through Montana to get an overall speedup for your trip of 1.67×?

Solution:

We know:

$$T_{old} = 25$$

$$S = 1.67$$

Then:

$$T_{new} = 25/1.67 = 15$$

So:

$$T_{Montana} = 15 - 1000/100 = 5$$

Finally:

$$v_{Montana} = 1500/5 = 300km/h$$

Practice Problem 1.2

A car manufacturing company has promised their customers that the next release of a new engine will show a 4× performance improvement. You have been assigned the task of delivering on that promise. You have determined that only 90% of the engine can be improved. How much (i.e., what value of k) would you need to improve this part to meet the overall performance target of the engine?

Solution:

We know:

$$\alpha = 0.9$$

$$S = 4$$

So:

$$k = 6$$

1.9.2 Concurrency and Parallelism

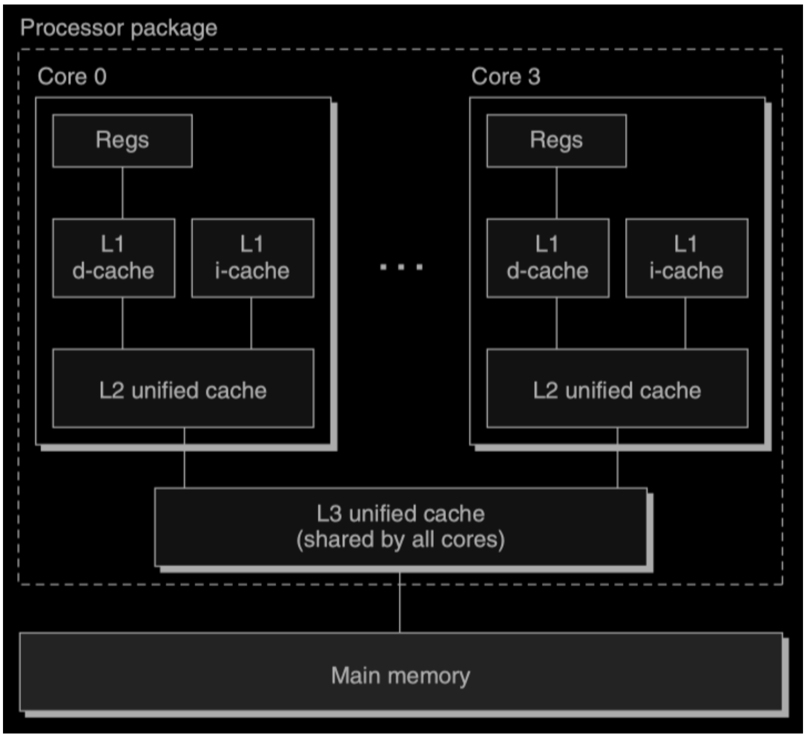
- Concurrency (并发)- the general concept of a system with multiple, simultaneous activities.
  - 多个程序可以同时运行的现象，重点在于它是一种现象；
  - 经过上下文快速切换，使得看上去多个进程同时都在运行的现象，是一种os欺骗用户的现象；
- Parallelism (并行) - the use of concurrency to make a system run faster.
  - 同一时刻可以多个进程在运行。

Thread-Level Concurrency

- With threads, we can have multiple control flows executing within a single process.
- 2 systems:
  - uniprocessor system - a system with a single processor
  - Multiprocessor system - a system with multiple processor
    - multi-core
    - hyperthreading

Multi-core

- Multi-core processors have several CPUs (referred to as "cores") integrated onto a single integrated-circuit chip.
- An example of 4 processor cores:



- The chip has four CPU cores, each with its own L1 and L2 caches, and with each L1 cache split into two parts.

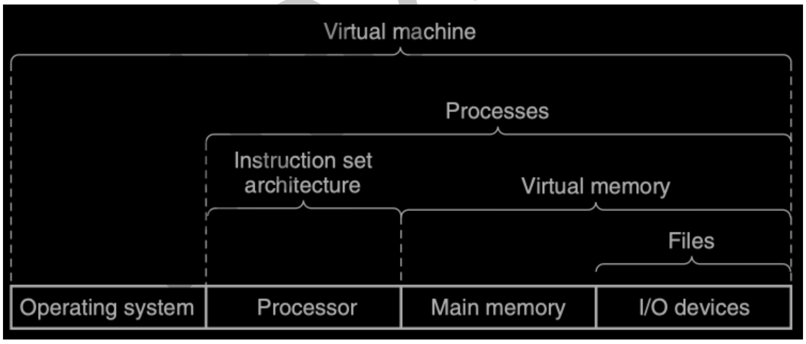
Hyperthreading

- Hyperthreading, sometimes called simultaneous multi-threading, is a technique that allows **a single CPU to execute multiple flows of control.**

Instruction-Level Parallelism

- At a much lower level of abstraction, modern processors can **execute multiple instructions at one time**, a property known as instruction-level parallelism.

1.9.3 The Importance of Abstractions in Computer Systems



- On the processor side, the instruction set architecture provides an abstraction of the actual processor hardware.
- On the operating system side, we have introduced three abstractions: **files as an abstraction of I/O devices, virtual memory as an abstraction of program memory, and processes as an abstraction of a running program.**
- The virtual machine, providing an abstraction of the entire computer, including the operating system, the processor, and the programs.