Welcome to CS/SE 2340

What do we mean by ‘computer architecture’

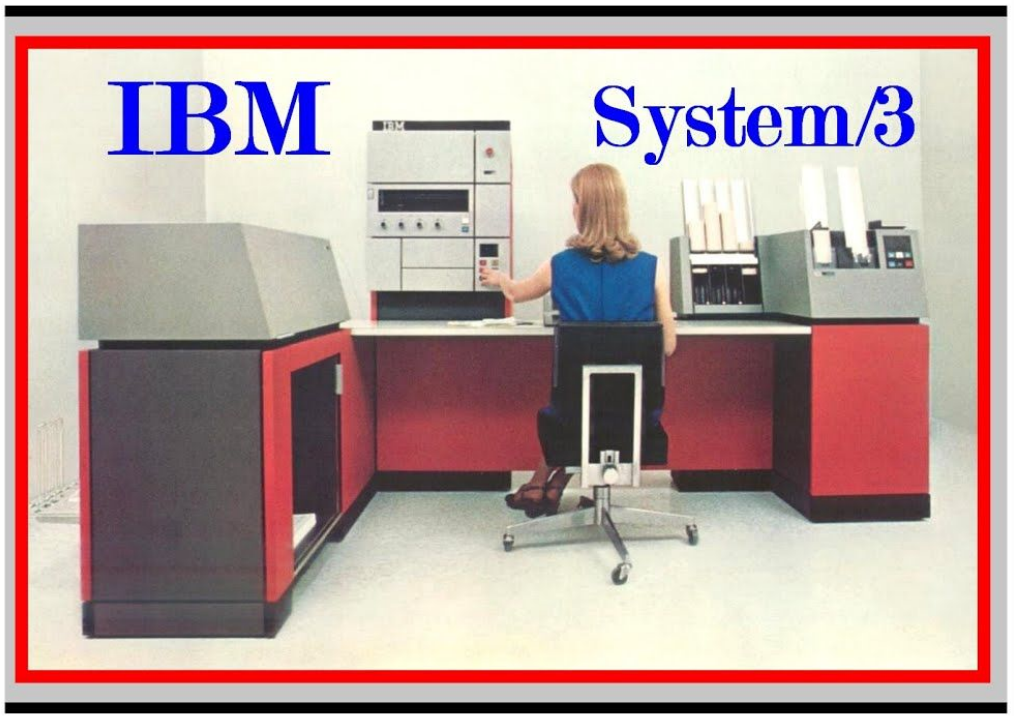
From mainframes to IoT

Mainframes

- early general-purpose computers

- still in use today by large organizations

Minicomputers



Personal Computers

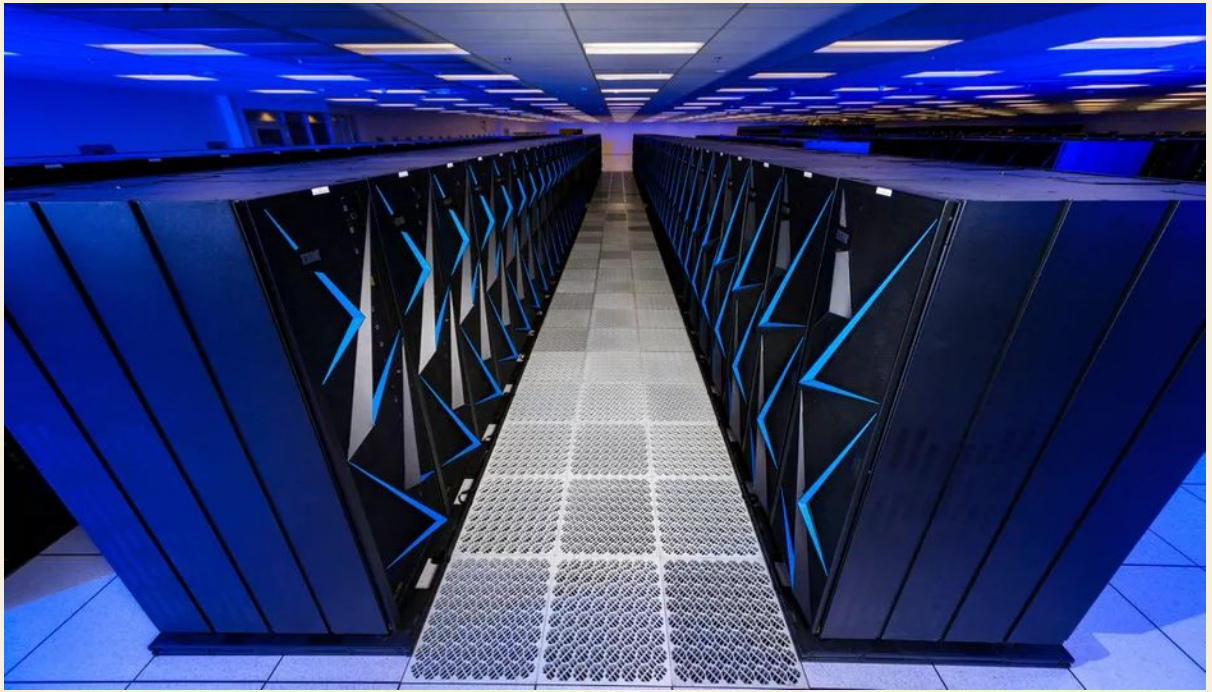
Personal mobile devices (hand-held computer)

Embedded systems (microcontrollers)

Some devices are connected to the Internet - IoT - the Internet of things

supercomputers

- 'exascale' computing

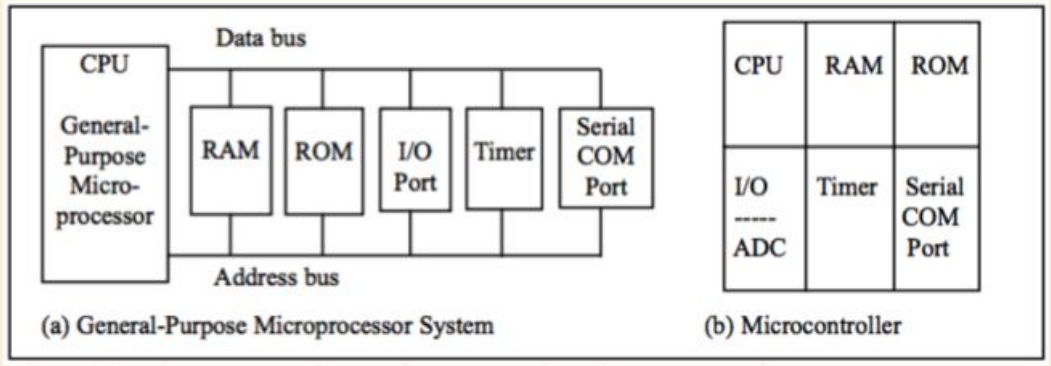


cloud computing

CPU - central processing unit

- microprocessor chip
  - connects to memory and peripherals via buses
  - used in PCs, servers

- microcontroller 'system on a chip'
  - memory, peripheral ports are internal
  - for embedded systems



similar design principles apply to microprocessors and microcontrollers

Memory

- x-byte
  - powers of 10
- x-bibyte
  - powers of 2

Decimal term	Abbreviation	Value	Binary term	Abbreviation	Value
kilobyte	KB	10^3	kibibyte	KiB	2^{10}
megabyte	MB	10^6	mebibyte	MiB	2^{20}
gigabyte	GB	10^9	gibibyte	GiB	2^{30}
terabyte	TB	10^{12}	tebibyte	TiB	2^{40}
petabyte	PB	10^{15}	pebibyte	PiB	2^{50}
exabyte	EB	10^{18}	exbibyte	EiB	2^{60}
zettabyte	ZB	10^{21}	zebibyte	ZiB	2^{70}
yottabyte	YB	10^{24}	yobibyte	YiB	2^{80}

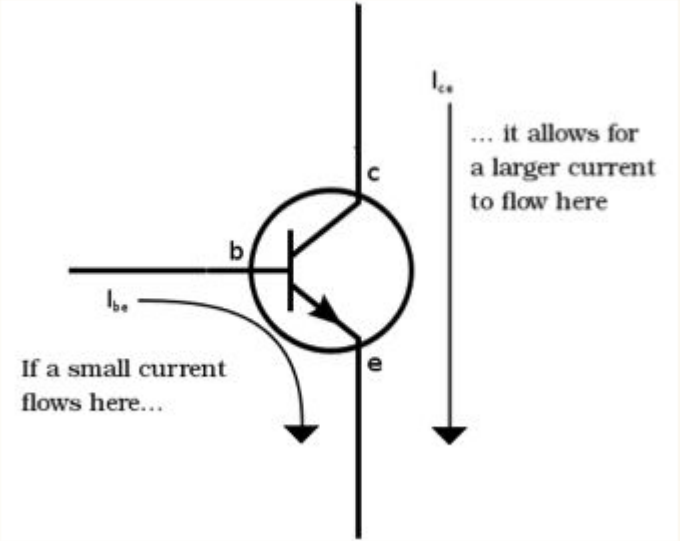
7 Great Ideas

Our Textbook

- Computer Organization and Design
- 6th edition - MIPS
- David Patterson
- John Hennessy

Design for Moore's Law

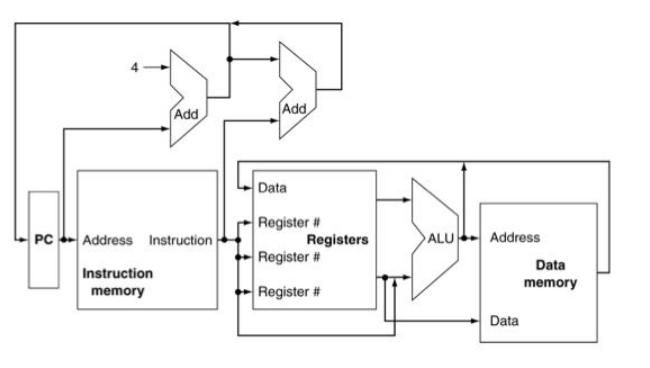
- 1965, Gordon Moore predicted that the number of transistors that would fit on chip would double every 1.5 to 2 years



1. Abstraction

Hardware abstraction

- system diagrams give the big picture of how components work together while omitting the details



Software abstraction

- int j = 5;

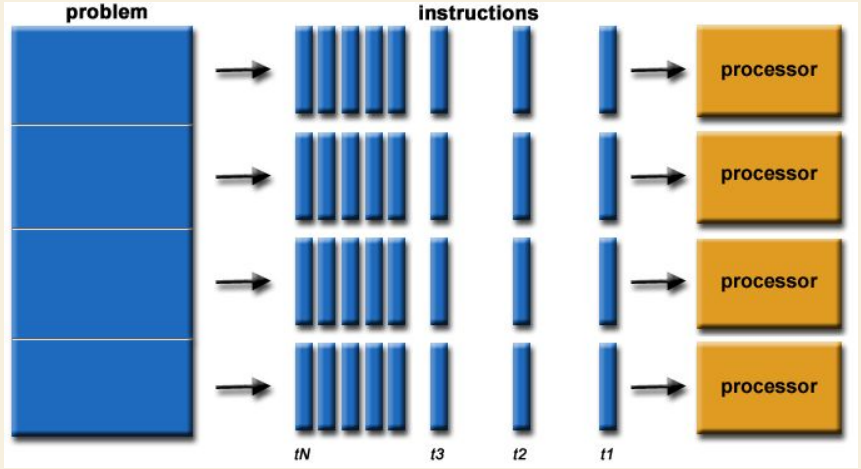
could be stored many ways:

```
0000 0101

0000 0000 0000 0101

0000 0000 0000 0000 0000 0101
```

2. Make the common case fast

- optimize the common case

3. Parallelism

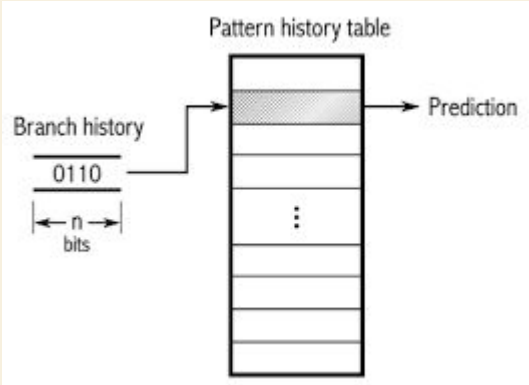


4. Pipelining

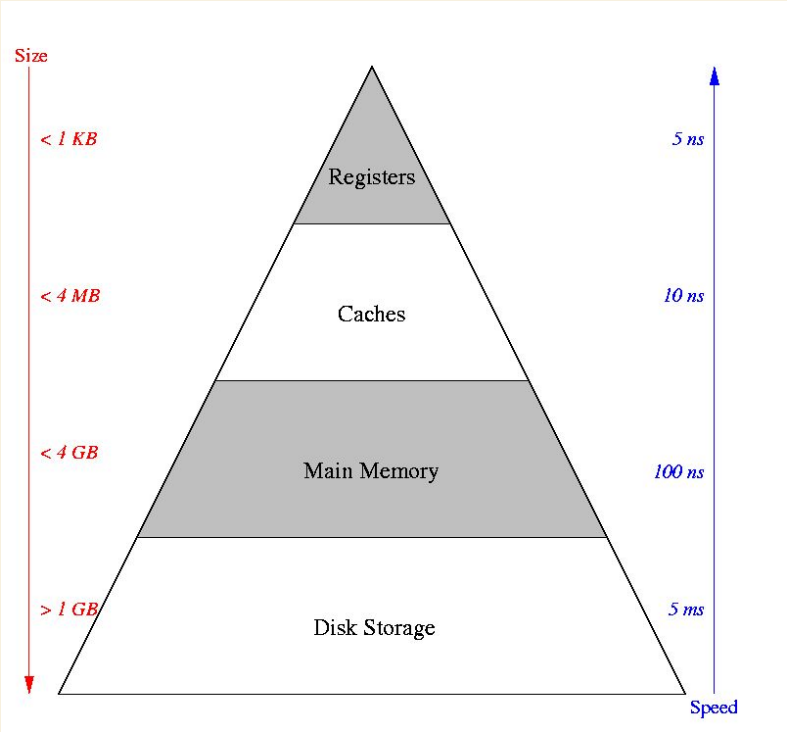
- using assembly line techniques for better throughput

5. Prediction

- rather than waiting to be sure, go ahead and predict which instruction to execute next



6. Memory hierarchy



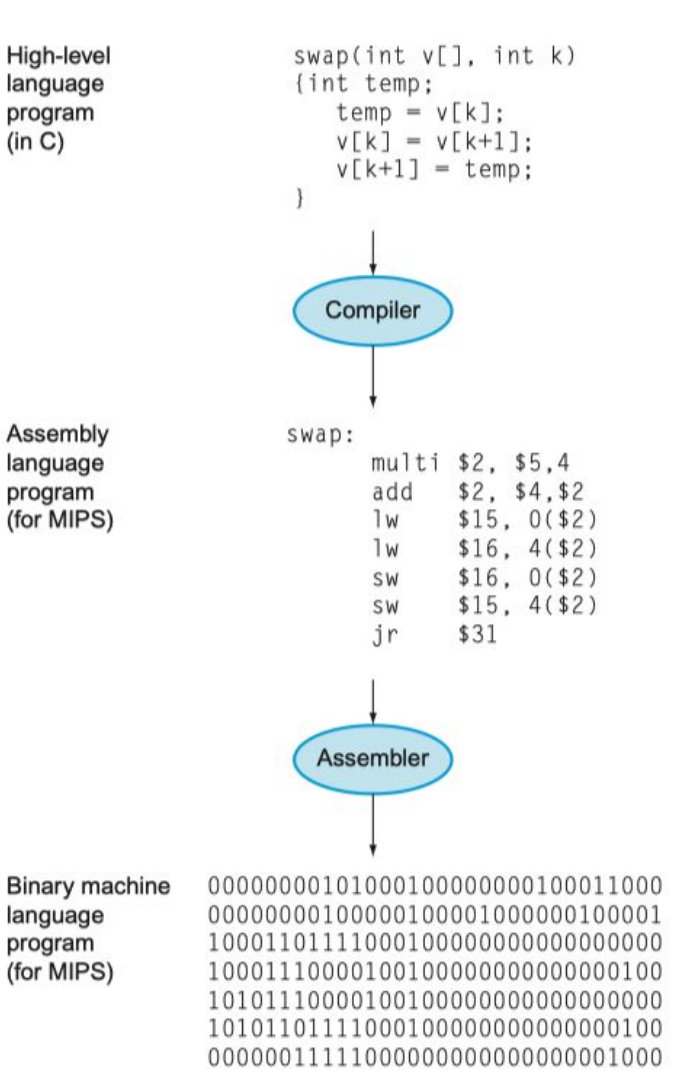
7. Redundancy

Under the Hood

Your programs

- high-level language compiled to assembly language, and then machine language
- or compiled directly to machine language

The ISA (instruction set architecture) is the set of instructions a CPU can execute



The Hardware

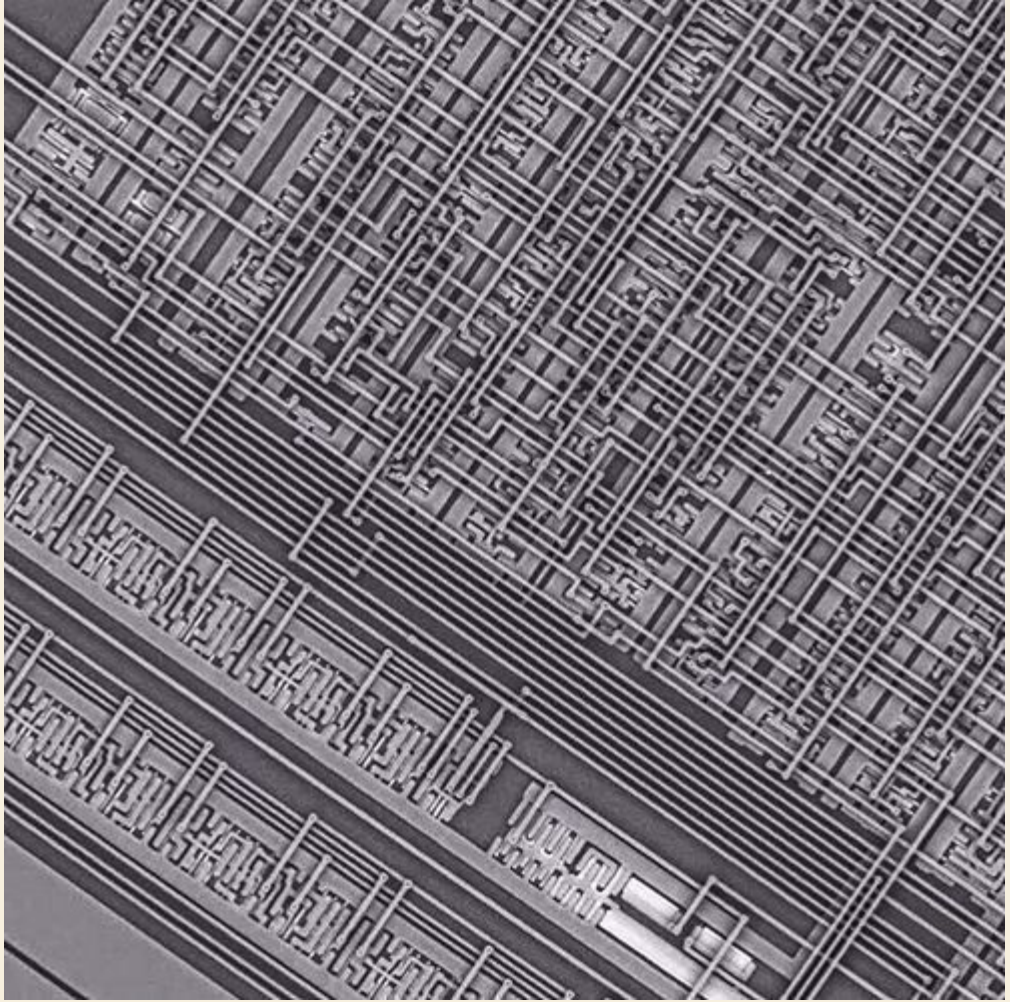
- chips connected on a board

Inside the A5

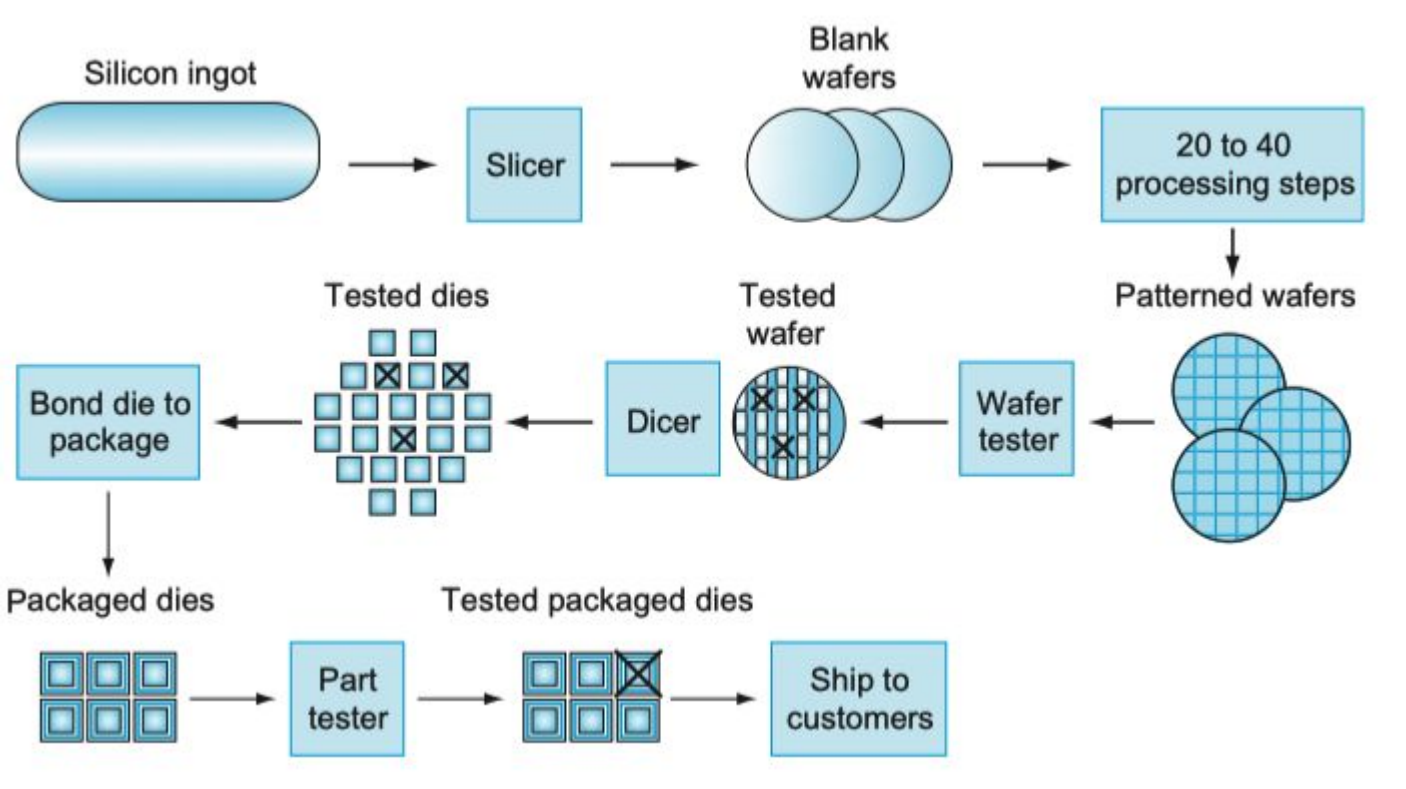
- dual-core ARM processor

Transistors

- viewed with an electron microscope



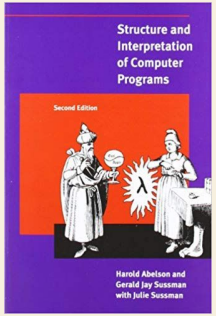
Chip manufacturing process



Welcome to the information revolution

information has a unique quality as a resource and a commodity, the utility of which, in combination with its other values, is so pervasive as to result in the now common appellation given to the period of history ahead as the 'information age'

- Encyclopedia of Library and Information Science, 1977



"Structure and Interpretation of Computer Programs"

- computer science is not a science and its significance has little to do with computers. The computer revolution is a revolution in the way we think and the way we express what we think.

Why study computer architecture

Why learn an assembly language?

- Does anyone still use assembly language?
  - Yes, for real-time and other system applications

- Why learn assembly language?
  - You will become a better programmer
  - You will finally understand what you are doing!!!
  - You will be able to write more efficient code

Quote

- Peter Norvig http://norvig.com/

- “Here at Google, sometimes we just throw stuff together,” Dr. Norvig said, during a meeting of the Google Trips team, in Mountain View, Calif. “But other times, if you’re serving billions of users, it’s important to do that efficiently. A 10-per-cent improvement in efficiency can work out to billions of dollars, and in order to get that last level of efficiency, you have to understand what’s going on all the way down.”