Computer Systems Research: Past and Future

- Butler Lampson

- People have been inventing new ideas in computer systems for nearly four decades, usually driven by Moore’s law. Many of them have been spectacularly successful: virtual memory, packet networks, objects, relational databases, and graphical user interfaces are a few examples. Other promising ideas have not worked out: capabilities, formal methods, distributed computing, and persistent objects. And the fate of some is still in doubt: parallel computing, RISC, and software reuse. The most important invention of the last decade, the World Wide Web, was not made by computer systems researchers. In the light of all this experience, I will talk about the topics that I think will be exciting to work on in the next few years.

Computer Systems Research

Butler Lampson

Microsoft

- The computer revolution has only just begun

- Outline
  - Context
  - History
  - Motivation
  - Challenges

Context: Moore's Law

- 100X performance / decade
- Will it last?
  - Room-temperature single-electron memory
  - Cell size 7 nm square
  - 1 cm^2 of these is 250 GB
  - Moore's law for 2018 predicts
    8MB x 10^4 = 80 GB

Moore’s Law: Qualitative Changes

- 100 processors/chip in 10 years
  - Or
    - Reconfigurable logic
    - 100K tiny processors
- RAM on-chip
- Chip-network bandwidth: 50 GB/sec

- (Long-distance latency 100 ms → 25 ms = no change)

Computer Science vs Engineering

- Science
  - Describe
  - Explain
- Engineering
  - Build

    “An engineer can do for a dime what any fool can do for a dollar.”
- Today mostly engineering
- Things take time

History: What Worked?

YES

Virtual memory*

Address spaces*

Packet nets*

Objects / subtypes

RDB and SQL

Transactions*

Bitmaps and GUIs*

Web

Algorithms

NO (Not Yet?)

Capabilities*

Fancy type systems*

Functional programming

Formal methods*

Software engineering

RPC (except for Web)*

Distributed computing*

Persistent objects

Security*

History: What Worked?

MAYBE

Parallelism

RISC

Garbage collection

Interfaces and specifications

Reuse

Works for	Unix filters
	Big things (OS, DB, browser)
Flaky for	Ole/COM

The Failure of Systems Research

- We didn’t invent the Web
- Why not? Too simple
  - Old idea
    - » But never tried
  - Wasteful
    - » But it’s fast enough
  - Flaky
    - » But it doesn’t have to work
- Denial: It doesn’t scale
  - Only from 100 to 100,000,000

What are Computers For?

- Simulation
- Communication among people
  - Storage = communication across time
- Control
  - Get physical
  - Get real (time)
  - Get mobile

Applications

- Simulation
  - Models of the real world (e.g., clothing, cities)
- Communication among people
  - Information at your fingertips
  - Telepresence
  - Home
- Control
  - Robots
  - MEMS

Motherhood Challenges

- Correctness
- Scaling
- Parallelism
- Reuse
- Trustworthiness
- Ease of use

Challenges: Programming

- Concurrency
  - Using 100 processors/chip
  - Matching biological concurrency
    - What can you do in 100 cycles?
- Declarative
  - SQL, spreadsheets the only successes so far
- Intelligence
  - Data models/class hierarchy/knowledge rep
- Uncertainty
  - Real-world input: speech, vision, ...
  - Adapting to environment

Challenges: Systems

- Systems that work
  - Meeting their specs
  - Always available
  - Adapting to changing environment
  - Evolving while they run
  - Made from unreliable components
  - Growing without practical limit
- Credible simulations or analysis
- Writing good specs
- Testing
- Performance
  - Understanding when it doesn’t matter

Challenges: Information

- Personal Memex
  - Remember everything a person hears or sees.
- Telepresence
  - Simulate being somewhere else
- Memex
  - Collect everything in the world’s libraries
  - Retrieve from it as well as a person can

Challenges: Physical World

- Seeing, hearing, speaking better than people
  - With real lighting, occlusion, noise, etc.
  - Recognizing speech and objects
  - In real time
- Self-organizing systems
  - Unreliable and changing components
  - Scale: molecules, cells, insects, cities
- Common sense
  - Requires good models of reality

Conclusions for Engineers

- Understand Moore’s law
- Aim for mass markets
  - Computers are everywhere
- Learn how to build systems that work