Introduction of CSDI

Rong Chen
Institute of Parallel and Distributed Systems
Shanghai Jiao Tong University
http://ipads.se.sjtu.edu.cn

Outline

Why Choose This Course?

Course information

How to do (System) Research

8 important problem in systems

Computer Systems: Huge Impact

Systems research has a high potential to directly advance technology

- Solving real problems in core components of system infrastructure
- Creating new system frameworks for low costs and high returns
- Building new systems that responds rapid demands in society

One more step after research prototypes can make a big difference

- Open source software: Linux, BSD, MySQL, Hadoop, BitTorrent, ...
- Commercial products: RISC CPU, GPU, Yahoo!, VMWare, ...

Source: Prof. Xiaodong Zhang

UC Berkeley: Most Influential Research

Theoretical Foundations and Influential Algorithms

- Fuzzy logic (Zadeh)
- Theory of NP-Completeness (Cook and Karp)
- Karmarkar's Algorithm for Linear Programming (Karmarka /Karp)
- Complexity of Cryptography (Blum, Micali and Goldwasser)

Advancing Technology and Improving Society Productivity

- Design automation: SPICE and others (Pederson, Rohrer, Sangiovanni, and Newton; -> Cadence and Synopsys)
- INGRES (Stonebraker and Wong; -> Oracle and PostgreSQL)
- Berkeley Unix (Fabry/Joy; -> Sun, FreeBSD, and NetBSD)
- IEEE 754 Standard for Floating-point Arithmetic (Kahan)
- RISC (Paterson/Sequin; -> chip designs in Sun, Fujitsu and others)
- RAID (Paterson/Katz/Gibson; -> more than 15 vendors)

Source: Prof. Xiaodong Zhang

Google's Key Enabling Technologies

MapReduce → Dataflow

BigTable → Spanner

GFS → Colossus

Is “simple” really simple?

A simple program?

```
#include <stdio.h>
int main() {
    printf(“hello world\n”);
    return 0;
}
```

How does it get executed?

How does it got translated to machine code?

How does the program get executed in detail?

Link, load, execute, finish?

Course Perspective

Our Course is Designer-Centric

- Purpose is to show that by knowing more about the underlying system, one can be more effective as a programmer

- Enable you to
  - Design reliable and efficient systems
  - Incorporate features that require hooks into systems
    - E.g., concurrency, signal handlers

- Cover material in this course that you won't see elsewhere

Course Topics

Non-volatile Memory

Crash Recovery

Multiprocessor: Concurrency and Locks

Transactional Memory/DB

NoSQL

Distributed Transactions

RDMA-enable Transaction Processing

RDMA-friendly Key/value Store

Low (Tail) Latency

Network Function Virtualization

Graph Query on RDF

Bugs

Outline

Why we study OS?

Course information

A tale of OS

OS structures

Faculty Information

Instructor

CHEN Rong 陈榕

Office: Room 3-402, Software Building

Contact:

13661816826 rongchen@sjtu.edu.cn

TA

SHEN Sijie 沈斯杰 ds_ssj@sjtu.edu.cn

Course Website

Main site:

http://ipads.se.sjtu.edu.cn/courses/csdi/

No textbook, a number of research papers

Work as a Team

Please send your group information to our TAs by 3.6

- Include team members

We will randomly assign teams if we didn't receive your emails

Will post the team information in the web-site by next Tuesday

Paper Presentation

- Each lecture will cover 3-4 papers
  - I will introduce some background knowledge regarding the papers
  - There will be 3-4 teams to present papers
  - TA will send scoresheet (or weixin) to students to grade the scores
    - I will grade the presentation as well

Paper Questions

- Each of you need to hand on your answers to the questions before the class
  - This ensures that you at least have scanned the first paper and thought about the questions

Nonverbal Skills	4 – Exceptional	3 – Admirable	2 – Acceptable	1 – Poor
Eye Contact	Holds attention of entire audience with the use of direct eye contact, seldom looking at notes or slides.	Consistent use of direct eye contact with audience, but still returns to notes.	Displayed minimal eye contact with audience, while reading mostly from notes.	No eye contact with audience, as entire report is read from note.
Body Language	Movements seem fluid and help the audience visualize.	Made movements or gestures that enhance articulation.	Very little movement or descriptive gestures.	No movement or descriptive gestures.
Poise	Displays relaxed, self-confident nature about self, with no-mistakes.	Makes minor mistakes, but quickly recovers from them; displays little or no tension.	Displays mild tension; has trouble recovering from mistakes.	Tension and nervousness is obvious; has trouble recovering from mistakes.

Verbal Skills	4 – Exceptional	3 – Admirable	2 – Acceptable	1 – Poor
Enthusiasm	Demonstrates a strong, positive feeling about topic during entire presentation	Occasionally shows positive feelings about topic	Shows some negativity toward topic presented.	Shows absolutely no interest in topic presented.
Speaking Skills	Uses a clear voice and speaks at a good pace so audience members can hear presentation. Does not read off slides.	Presenter's voice is clear. The pace is a little slow or fast at times. Most audience members can hear presentation.	Presenter's voice is low. The pace is much too rapid/slow. Audience members have difficulty hearing presentation.	Presenter mumbles, talks very fast, and speaks too quietly for a majority of students to hear & understand.

Timing	4 – Exceptional	3 – Admirable	2 – Acceptable	1 – Poor
Length of Presentation	Within two minutes of allotted time +/-.	Within four minutes of allotted time +/-.	Within six minutes of allotted time +/-	Too long or too short; ten or more minutes above or below allotted time.

Content	4 – Exceptional	3 – Admirable	2 – Acceptable	1 – Poor
Subject Knowledge	An abundance of material clearly related to the research is presented. Points are clearly made and evidence is used to support claims	Sufficient information with many good points made, uneven balance and little consistency.	There is a great deal of information that is not clearly integrated or connected to the research.	Goal of research unclear, information included that does not support research claims in any way.
Organization	Information is presented in a logical and interesting sequence which audience can follow. Flows well.	Information is presented in logical sequence which audience can follow.	Audience has difficulty following presentation because the presentation jumps around and lacks clear transitions.	Audience cannot understand presentation because there is no sequence of information.
Visuals	Excellent visuals that are tied into the overall story of the research.	Appropriate visuals are used and explained by the speaker.	Visuals are used but not explained or put in context.	Little or no visuals, too much text on slides.
Mechanics	Presentation has no misspellings or grammatical errors.	Presentation has no more than two misspellings and/or grammatical errors.	Presentation has three misspellings and/or grammatical errors.	Presentation has many spelling and/or grammatical errors.

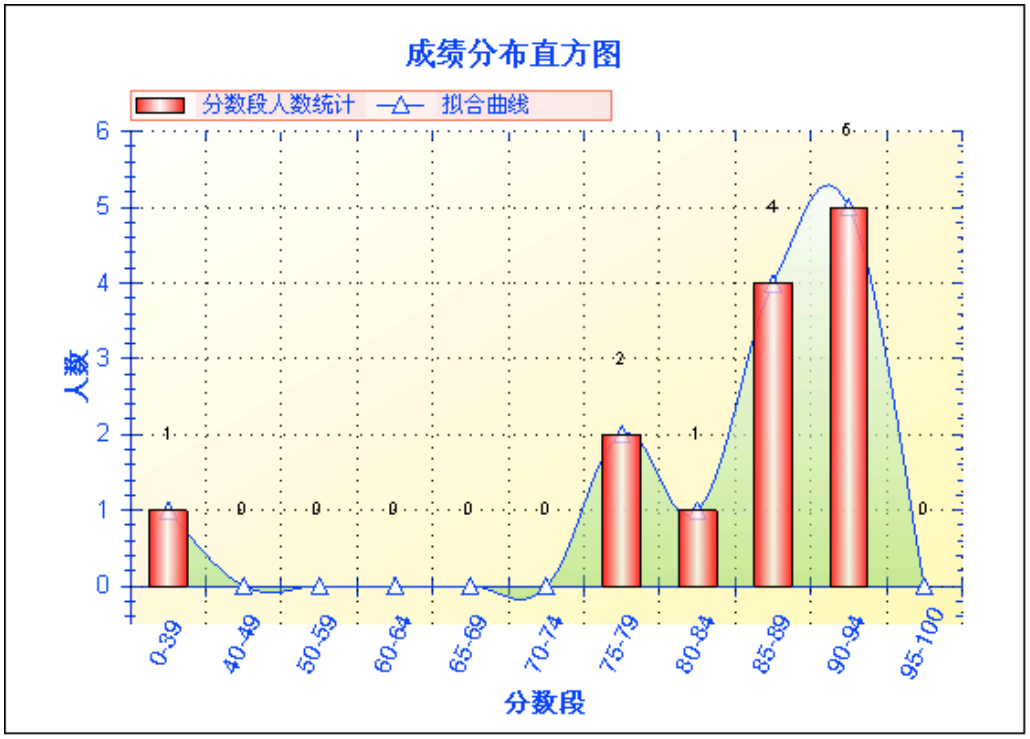
Source: http://pages.cs.wisc.edu/~david/courses/cs758/Fall2010/handouts/Oral_presentation_grading.pdf

Grading

General (Tentative)

- Final Exam：60%
  - Papers will be reflected in exam
- Paper reading, Q&A: 10%
  - Help you understand systems
- Paper presentation: 25%
- Course performance 5%
  - Q&A in classroom

Haibo: I am a nice man☺



备注	成绩记录均采用四级分值，分十个数级，即：									
分数级	A+	A	A-	B+	B	B-	C+	C	C-	D
折算百分数	96~100	90~95	85~89	80~84	75~79	70~74	67~69	63~66	60~62	0~59
各分数段比例	20.24%		70.24%			9.52%				
学校要求比例	<=20%		--			>=10%				

Rong: I am also a nice man☺

备注	成绩记录采用A+至F的十一级记分制，或者“通过/不通过”（2016年9月及以后入学的研究生适用）												
	百分数	95~100	90~94	85~89	82~84	78~81	75~77	71~74	67~70	63~66	60~62	<60	通过	不通过
	分数等级	A+	A	A-	B+	B	B-	C+	C	C-	D	F	P	F
	各分数段比例	4.76%	16.67%	23.81%	23.81%	19.05%	0%	2.38%	2.38%	2.38%	0%	4.76%	95.24%	4.76%
	学校建议比例	<=5%	--						>=10%				--	

备注	成绩记录均采用四级分值，分十个数级，即：									
	分数级	A+	A	A-	B+	B	B-	C+	C	C-	D
	折算百分数	96~100	90~95	85~89	80~84	75~79	70~74	67~69	63~66	60~62	0~59
	各分数段比例	20.24%		70.24%			9.52%				
	学校要求比例	<=20%		--			>=10%				

Computer System Research

Computer Systems research: not a science

Science (results research)

- Measure and quantitate existing (natural) phenomenon
- Building models to understand the world

Engineering (idea research)

- Building useful systems and tools

Computer systems research: combining the two

Computer Systems research: not a science

Not physical law limitation

- Usually depends on your own capability: lots of systems heroes (e.g., last 90s)
  - UNIX, Linux, Mac, Database
- There usually only a few core developers in many companies
  - MapReduce, Google Cluster, BigTable

Great flexibility

- Upside: flexibly define new problem to solve
- Downside: may easily invent irrelevant wheel

Tradeoff is usually a key

- Rarely free lunch available

Problem/Phenomenon Driven Ideas

Phenomenon: programs exhibit locality

- Idea: cache

Phenomenon: most programs are simple

- Idea: RISC

Phenomenon: social-network is self-similar

- Idea: ?

How to Choose a Research Problem？

Have some religion about your idea or result

- Will greatly help when the going gets tough

Don’t worry about people stealing your ideas

- Feedback from sharing >> cost of theft
- Getting people interested in your idea will be much harder than getting them to “steal” it.

Most ideas are dead ends, few endure

- learn to discard bad ideas quickly
- learn to recognize a great ideas

Slides from R.P. Martin

Top Conferences in Systems

Most Reputed

- **SOSP**: ACM Symposium on Operating Systems Principles
- **OSDI**: Usenix Symposium on Operating System Design and Implementation

Top Conferences

- **EuroSys**: European Conference on Computer Systems
- **Usenix ATC**: Usenix Annual Technical Conference

Sub-area Top Conferences

- **NSDI**: Usenix conference Networked System Design and Implementation
- **FAST**: Usenix conference on File and storage technologies
- **Usenix Security**: Usenix Security Symposium

Top Computer Architecture Conferences

ISCA: International conference on computer architecture

MICRO: International Symposium on Microarchitecture

HPCA: International conference on High-performance Computer Architecture

ASPLOS: International conference on Architectural support for programming language and operating systems

Key You Refreshed

Follow top conferences

- SOSP/OSDI, EuroSys, Usenix ATC, FAST

Read trend in industry

- lwn.net, infoworld, slashdot.com,

Learn who are the leaders in your field, know what they are doing

- But be critical if you want to follow…

Investigate

Has your idea been done before?

know what are in classic papers

Why is your idea “better”?

Why will your result be important?

Who will care?

Final impact if you’re successful?

Explore your idea

3 Approaches to systems research:

- Build a prototype
  - To do right is very hard
  - Ultimate validation
- Build a simulation
  - Not as hard but is it credible?
- Build a measurement apparatus

A sufficient analysis background is critical to all 3 approaches

Evaluate your idea

How is your idea better? Result novel?

Measure it

- Latency, throughout, fault tolerance
- space (still an issue?)
- usability, manageability (new!)

Judgment on artistic merit

- Is your result or idea exciting?
  E.g. Cray-1, Unix, Risc, Fortran, self-similarity

One Tip: Incommensurate Scaling

Scientific method vs. computer scientific method

Scientific method

- Control 1 parameter at a time, observe results

Computer scientific method

- Change everything
- If data doesn't fit your intuition, throw it away!

No magic

What if it does not work?

No magic, everything can be figured out

Form a hypothesis

- Cross-check with other evidence
- test with a simple experiment

Find who’s done it before or built it and ask them

Newer evaluation points

How will your idea mesh with the installed base?

- Huge deployment costs?

What are the switching costs over the current or obvious solutions?

- Is your idea 10x better than today?

How will predictable technology advances impact your idea?

- Will your idea be 10x better in 5 years?

Choose a research topic?

Be precise

Is security a research topic?

- How about program analysis?

Find a Research Topic

Pebbles on the beach

Hammer- and - nails

Be Super-Critical

Be super-critical on being convinced

- Find the real problems

Past Lessons

- Spent half a year working on a weak problem
- Finally gave up and no result

Stick on It when being convinced

Stick on the problem and believe it

I saw some of my students spent lots of time bouncing between diverse topics

Learn something

How to judge whether we fail or not?

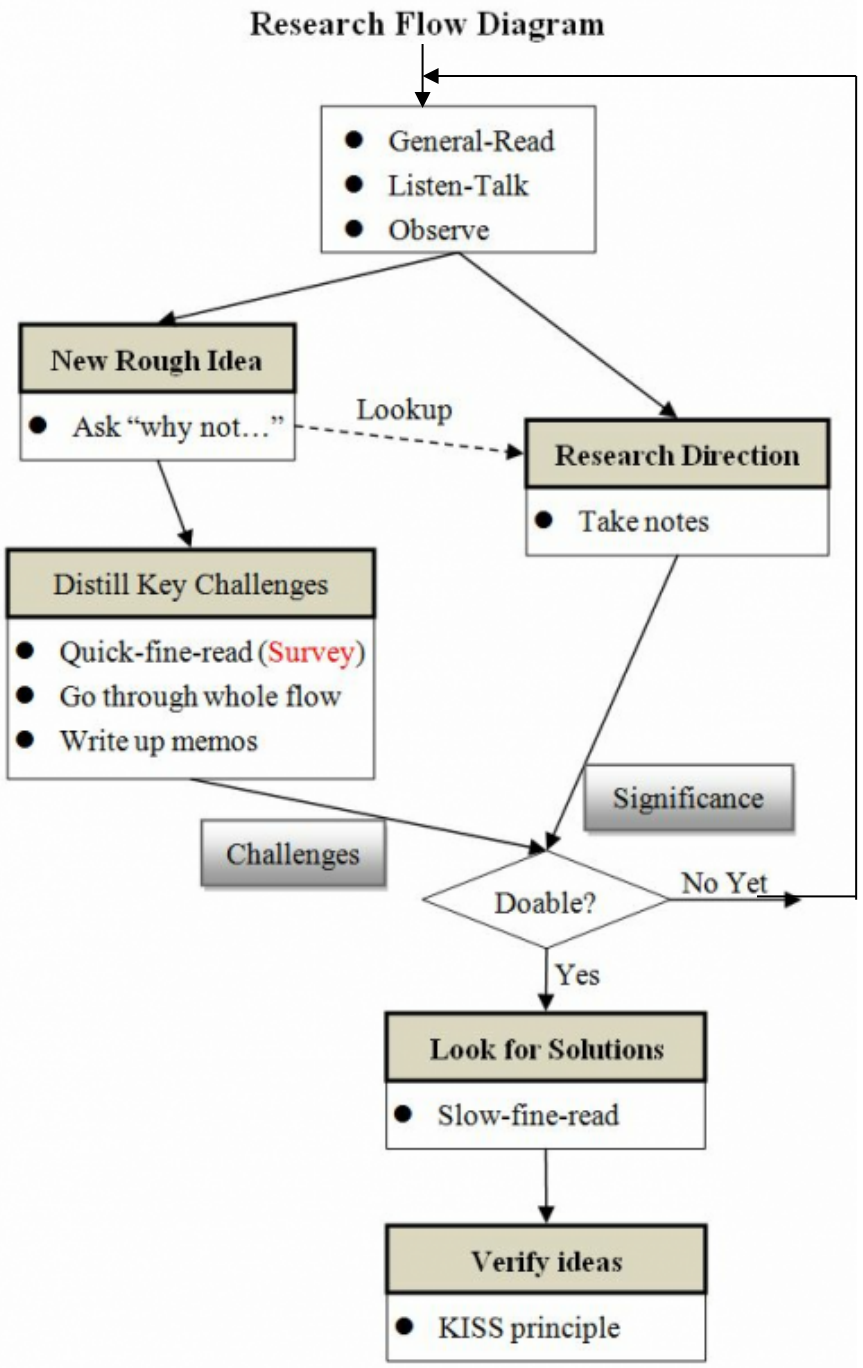
Criteria

- Whether we do have a deep understanding of this targeted research topic

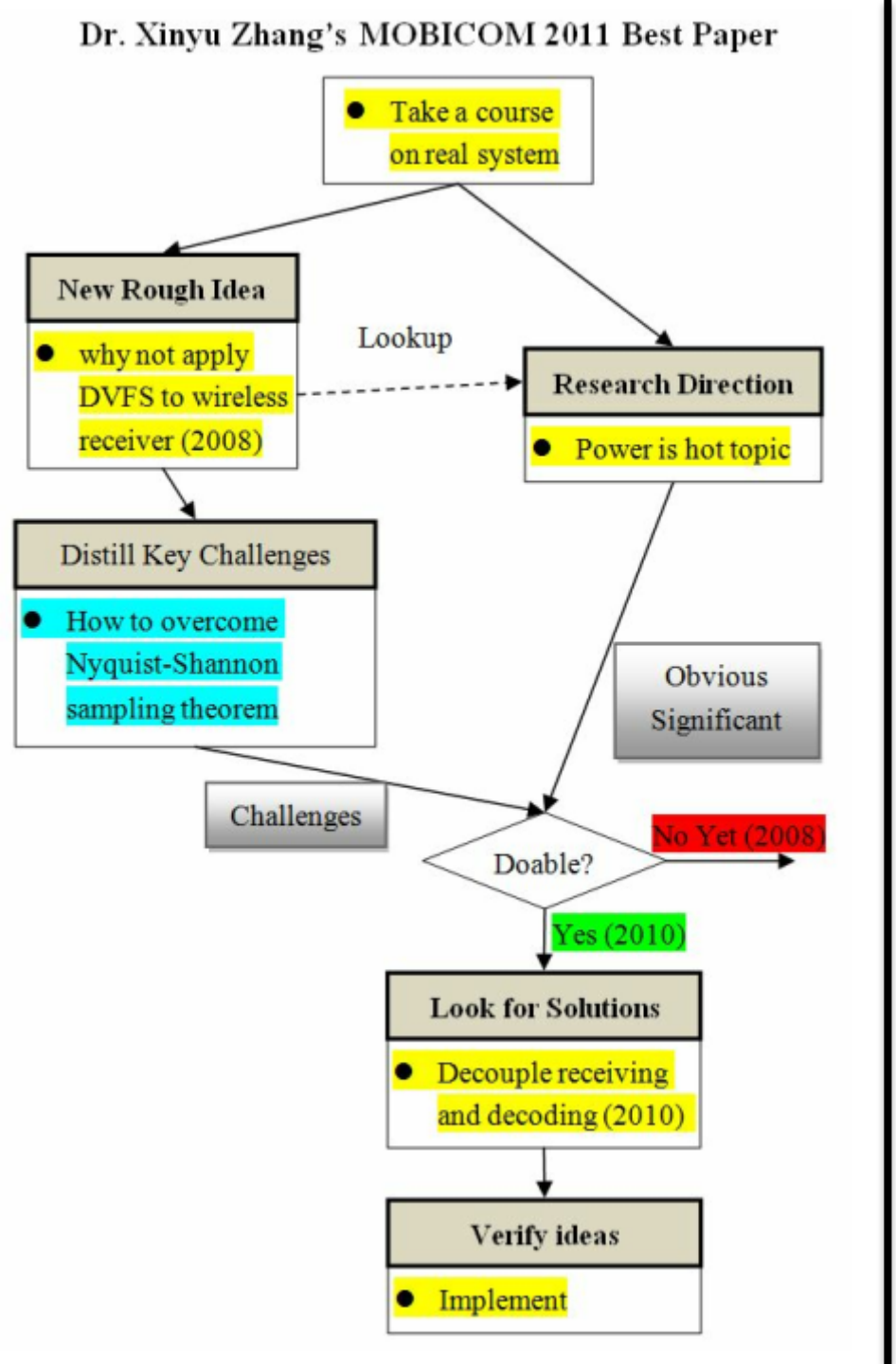
Rules in Research?

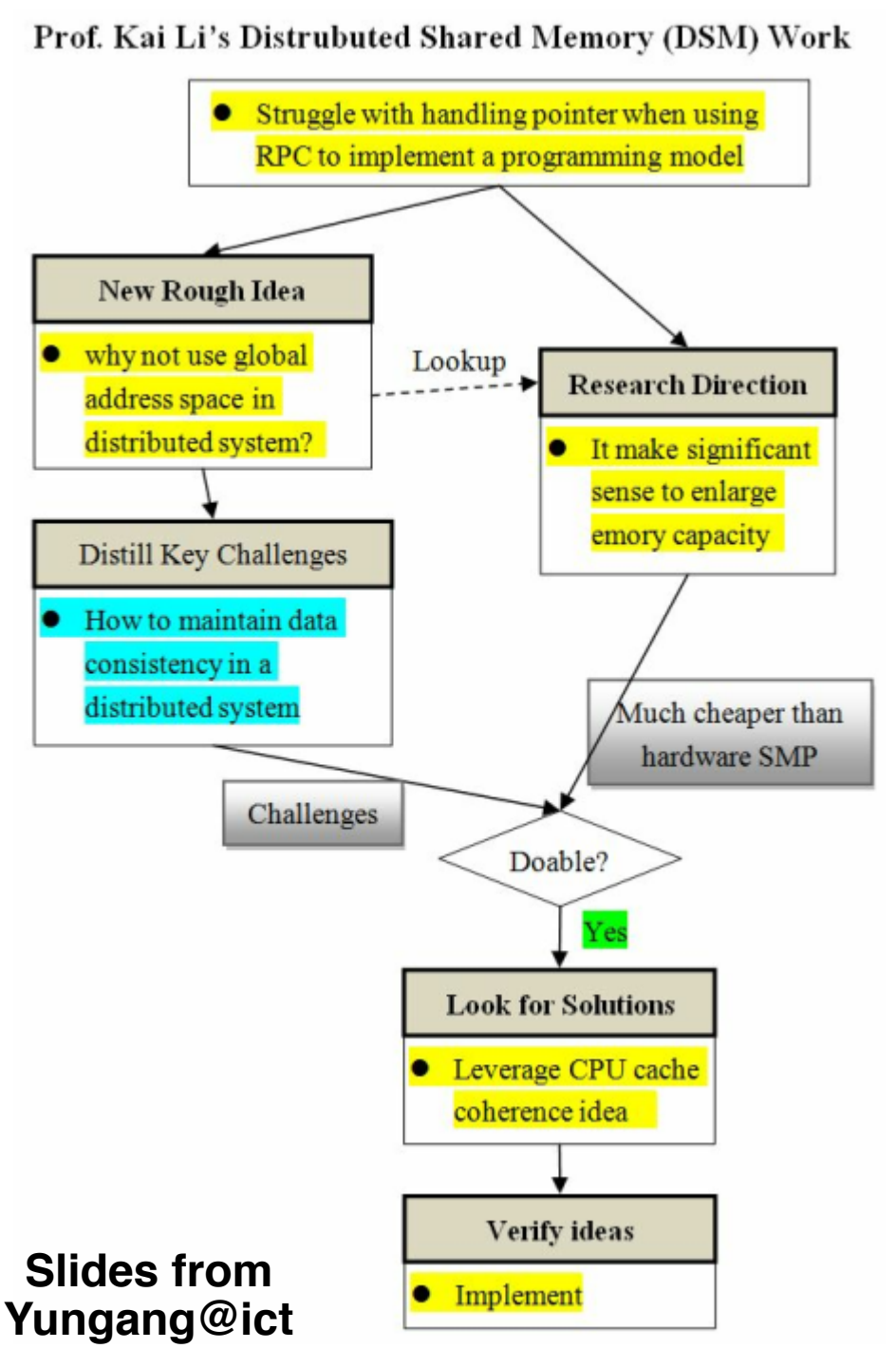
Prof. YY Zhou (UCSD)

- **Phase 0:** Research direction Identification
- **Phase 1:** Problem space understanding (1-2 months)
- **Phase 2:** Solution Seeking (1-2 months or more)
- **Phase 3:** Feasibility Study (1~2 months)
- **Phase 4:** If feasible, thoughts synchronization (2-4 weeks)
- **Phase 5:** System and experiment design (2-3 weeks)
- **Phase 6:** Implementation (3-6 months)
- **Phase 7:** Result Collection
- **Phase 8:** Proof read papers



Slides from Yungang@ict


Dr. Xinyu Zhang's MOBICOM 2011 Best Paper
Take a course on real system
New Rough Idea
why not apply DVFS to wireless receiver (2008)
Lookup
Research Direction
Power is hot topic
Distill Key Challenges
How to overcome Nyquist-Shannon sampling theorem
Obvious Significant
Challenges
Doable?
No Yet (2008)
Yes (2010)
Look for Solutions
Decouple receiving and decoding (2010)
Verify ideas
Implement


Prof. Kai Li's Distrubuted Shared Memory (DSM) Work
Struggle with handling pointer when using RPC to implement a programming model
New Rough Idea
why not use global address space in distributed system?
Lookup
Research Direction
It make significant sense to enlarge emory capacity
Distill Key Challenges
How to maintain data consistency in a distributed system
Much cheaper than hardware SMP
Challenges
Doable?
Yes
Look for Solutions
Leverage CPU cache coherence idea
Verify ideas
Implement
Slides from Yungang@ict

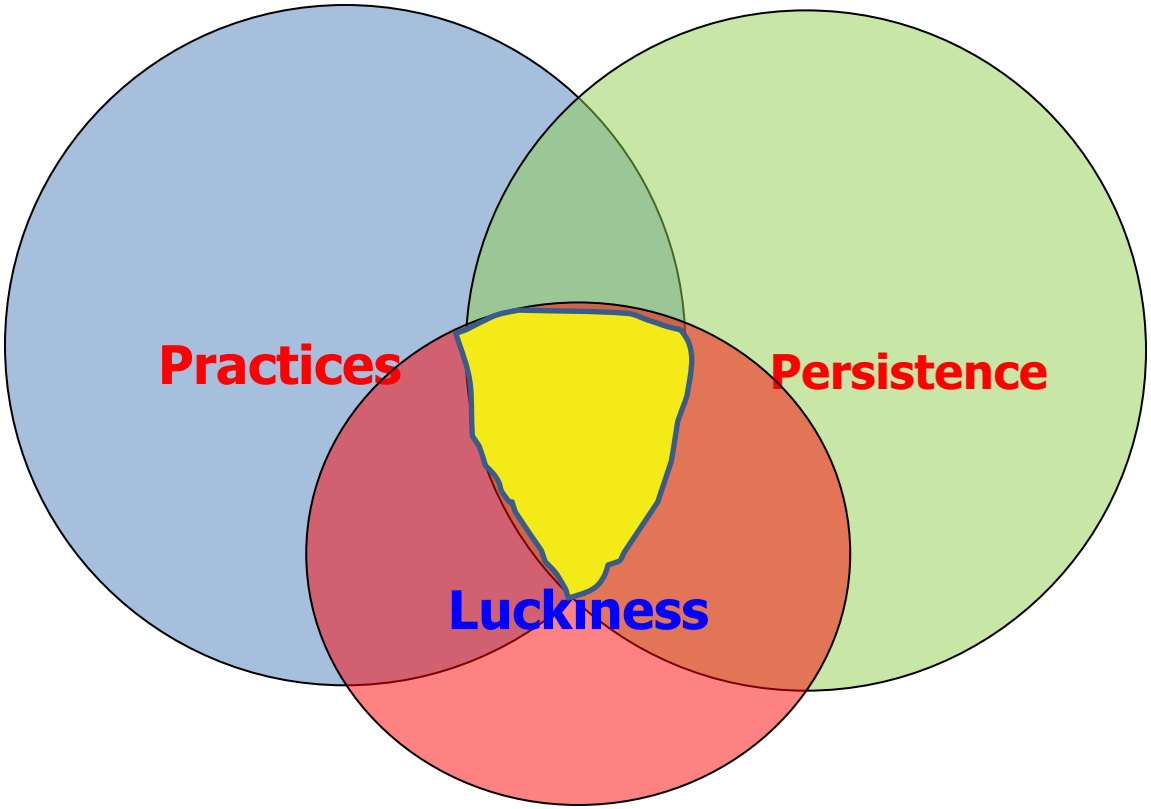
How to push ahead projects?

- **KISS: Keep It Simple, Stupid!**
  - ~~Pitfalls: make every step perfect！Perfectness kills us~~
  - **Move forward fast, evolve eventually**

这里是美国V：【先完成 后完美】美国工程院院士、Google Fellow辛格的哲学：先用个简单方案解决80%的问题，再慢慢解决剩下20%。许多优秀人才都败在这一点：一开始追求完美，费时耗力，最后不了了之。而针对剩下20%问题的每项改进都要说清楚理由，否则即使有效也不会采用，因为将来可能是个隐患。via@商界智库

Slides from Yungang@ict

What else in research?



Slides from
Yungang@ict

amplab

A Berkeley View of Big Data

- Mesos：Rejected for 2 years before accepted by NSDI'11
- Spark：Rejected for 2 years before accepted by NSDI'12

Sparrow: Distributed, Low Latency Scheduling

Kay Ousterhout, Patrick Wendell, Matei Zaharia, Ion Stoica
University of California, Berkeley

11 Acknowledgments

shape the final version of the paper. Finally, we thank the reviewers from HotCloud 2012, OSDI 2012, NSDI 2013, and SOSP 2013 for their helpful feedback.

Slides from Yungang@ict

Welcome to Research

8 IMPORTANT PROBLEMS IN OPERATING SYSTEMS

8 Important Problems (Not a Rank)

Scale Up

Security & Privacy

Energy Efficiency

Mobility

Write Correct (Parallel) Code

Scale Out

Non-Volatile Storage

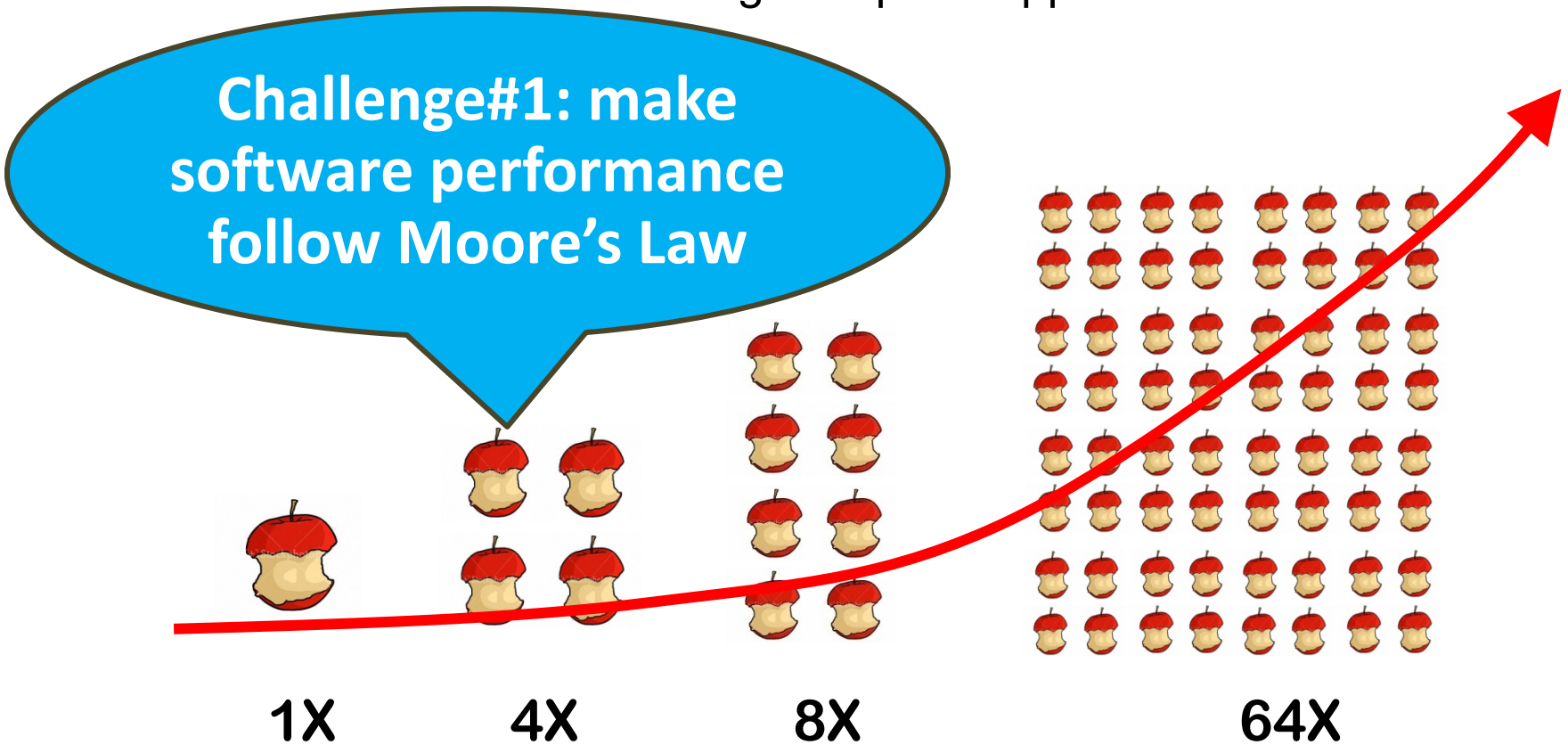
Virtualization

#1: Scale up (or Performance Scalability)

Multicore Evolution

Multicore is commercially prevalent recently

- Eight cores and Twelve cores on a chip are common
- Hundreds of cores on a single chip will appear in near feature



Why (Operating) Systems Matter?

Terms of (operating) system here

- Broadly defined, including hypervisor, operating systems, runtime environments

(Operating) systems manages and hide resources from applications

- Hide tedious and complex low-level details
- Provide execution environments to apps

(Operating) systems determines app perf. in many cases

- Many apps spend a large fraction of time in system software

Operating System Meets Multicore

(Operating) systems 20 years ago

- Enjoys the free lunch provided by hardware (Moore's Law)
- Recall the Andy-Bill's Law

Multicore: ending the free lunch!

- Software must evolve with hardware changes

OS: a vital role to bridge the gap between apps and hardware

- Need to evolve itself with multicore
- Need to evolve for apps with multicore

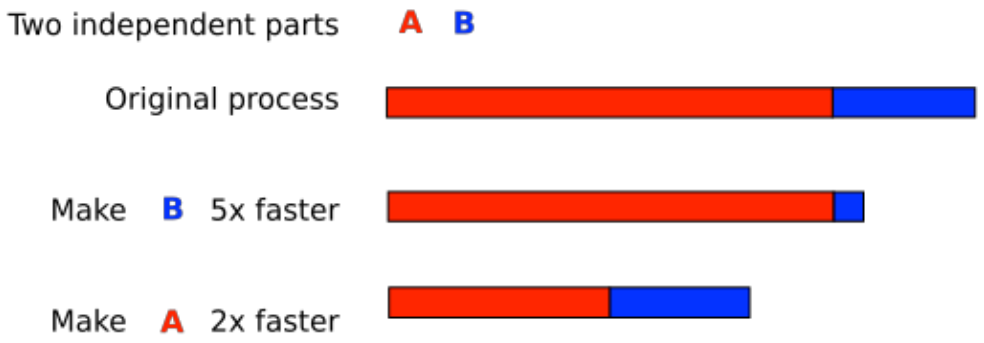
What is scalability?

Application does N times as much work on N cores as it could on 1 core

Scalability may be limited by Amdahl's Law:

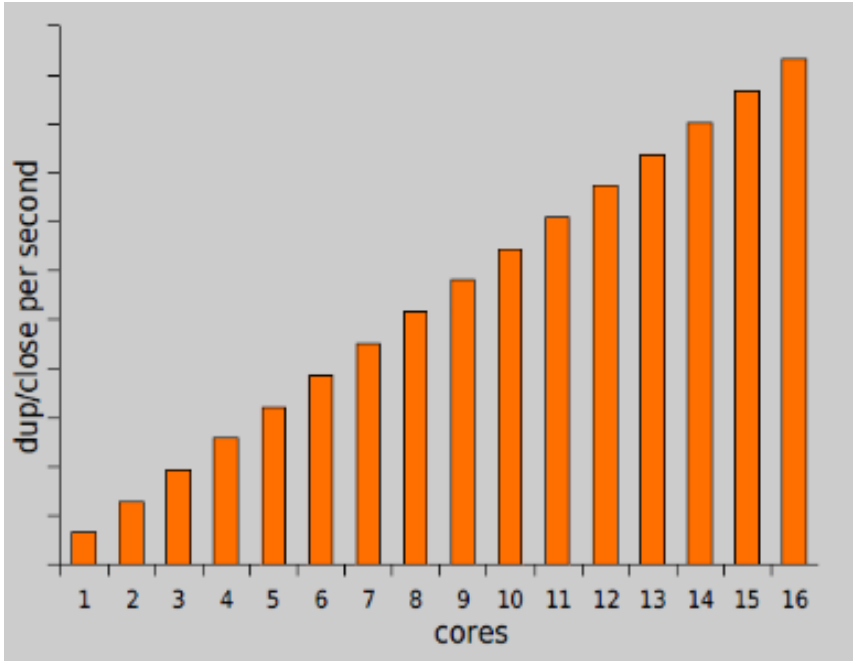
- Locks, shared data structures, ... Shared hardware (DRAM, NIC, ...)

$$\frac{1}{(1-P) + \frac{P}{S}}$$

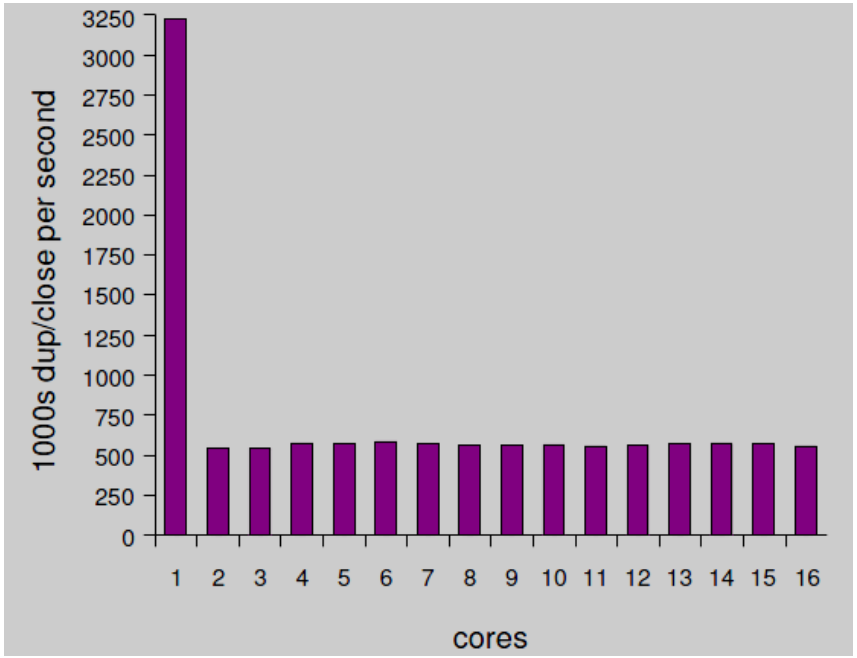


Motivating example: file descriptors

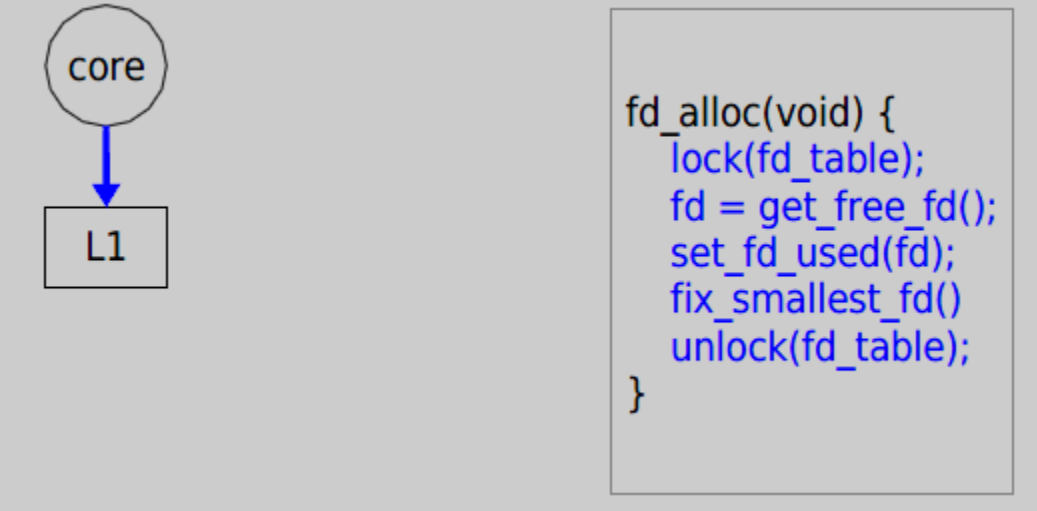
Ideal FD performance graph



Actual FD performance

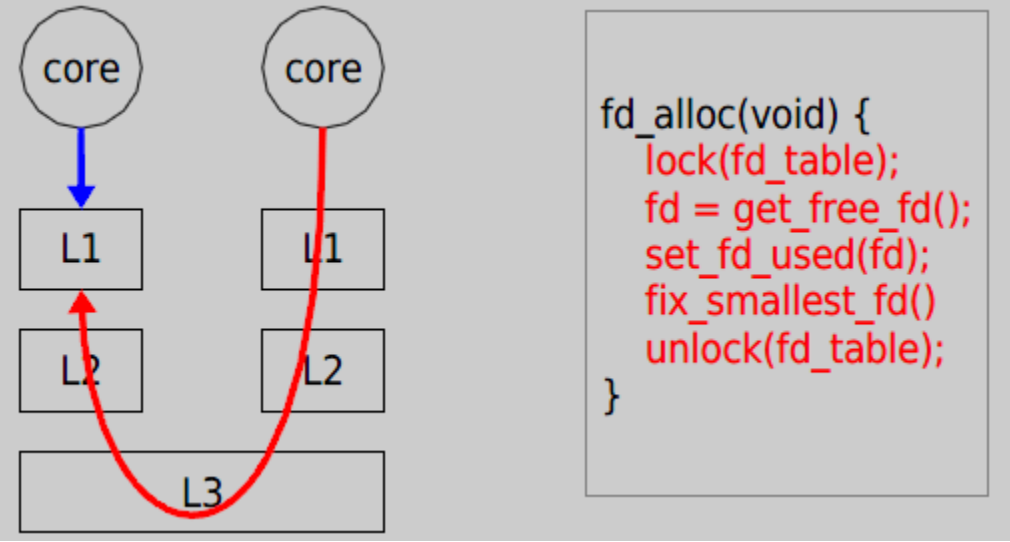


Why throughput drops?



Load fd_table data from L1 in 3 cycles.

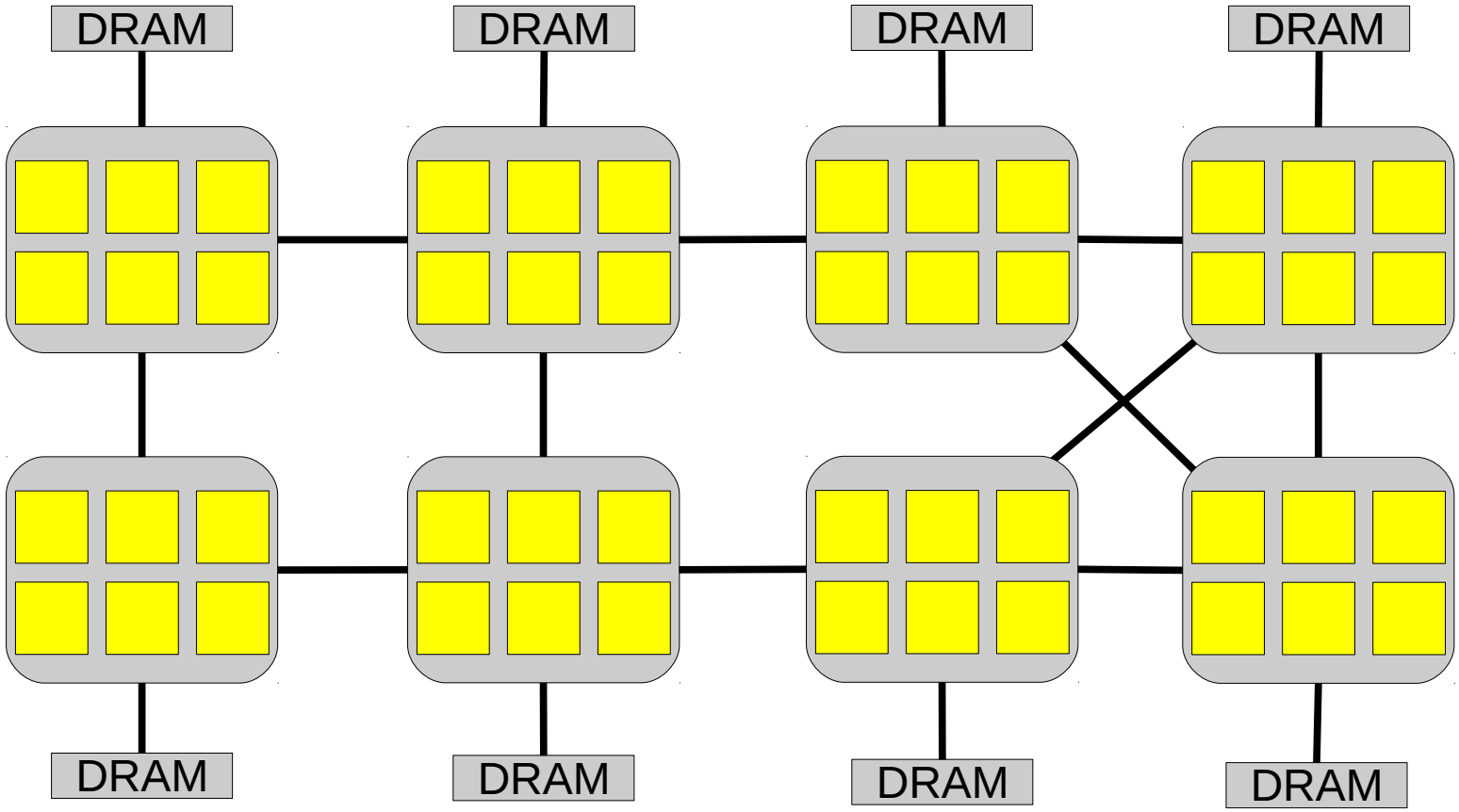
Why throughput drops?



Now it takes 121 cycles!

Off-the-shelf 48-core server

6 core x 8 chip AMD



Scale Up OS

Better abstraction to user applications

Eliminating non-scalable synchronization

Minimize sharing of common data structures

You will fill this..

Issue#2: Security & Privacy

Security

From Wikipedia:

> Security is the degree of resistance to, or protection from, harm

Why Operating System matters?

- Operating system (including hypervisors) lies in the lowest levels in a computer
- How could you protect your application & data without properly protect your OS

Easy Approach: Phishing

“hey! check out this funny blog about you...”

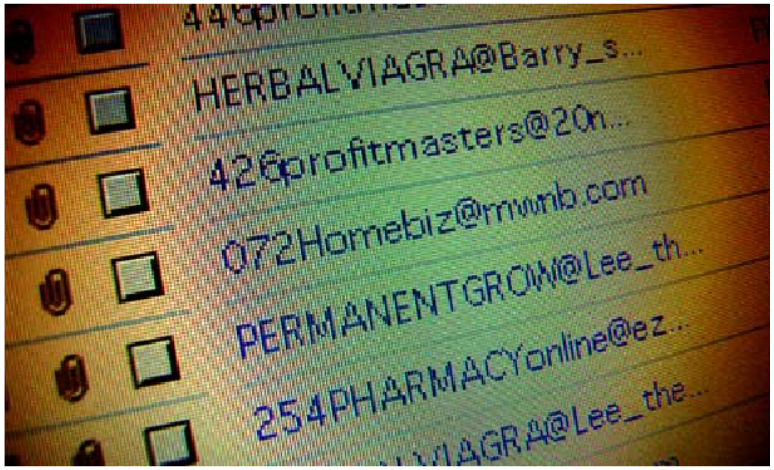
guardian.co.uk

News | Sport | Comment | Culture | Business | Money | Life & style

News › Technology › Microsoft

Hotmail password breach blamed on phishing attack

Bobbie Johnson, San Francisco
guardian.co.uk, Tuesday 6 October 2009 07.58 BST
Article history



Attack: Spam emails may have been responsible. Photograph: Roger Tooth

Microsoft has confirmed that the publication of thousands of Hotmail passwords was the result of a phishing attack against users of the popular email service.

Precise details of the strike, which was first uncovered on Monday, remain unclear. But in a statement, the American software company said

Sophisticated Approaches: Botnets and Malware

GSN. Government Security News

Home | Technology Sectors | Market Sectors | Buyer's Guide | Back Issues | Videos

Technology Sectors

- Access Control | Identification
- CBRNE | Detection
- Communications
- Cyber Security
- Disaster Preparedness | Emergency Response
- Education | Training
- IT Security
- Perimeter Protection
- Video Surveillance | CCTV

Market Sectors

- Airport | Aviation Security
- Border Security
- Federal | Agencies | Legislative
- Infrastructure Protection
- Law Enforcement | First Responders
- Maritime | Port Security
- Military | Force Protection
- State | Local Security
- Security Services

Treasury Dept. has cloud hacked

Mon, 2010-05-10 02:20 PM
By: Melissa Jane Kronfeld

The Treasury Department was hacked last week, leaving the Web site for its Bureau of Engraving and Printing - the agency responsible for printing U.S. dollars - down from May 3 to May 7.

The Treasury had moved to a cloud platform last year and the department blamed its cloud computing provider (the Treasury's Web site is hosted by Network Solutions) for the incident.

In a statement released May 4, the Treasury Department said, "The Bureau of Engraving and Printing (BEP) entered the cloud computing arena last year. The hosting company used by BEP had an intrusion and as a result of that intrusion, numerous websites (BEP and non-BEP) were affected. On May 3, the Treasury Government Security Operations Center was made aware of the problem and subsequently notified BEP.

"BEP has four Internet address URLs all pointing to one public website. Those URLs are BEP.gov; BEP.treas.gov; Moneyfactory.gov and Moneyfactory.com. BEP has since suspended the website. Through discussions with the provider, BEP is aware of the remediation steps required to restore the site and is currently working toward resolution."

Roger Thompson, chief research officer for IT security software vendor AVG Technologies USA, Inc. of Chelmsford, MA, was the first to notice the hack, and reported it the FBI. Thompson revealed that the hackers had added a tiny snippet of a virtually undetectable iframe HTML code that redirected visitors to a Ukrainian Web site. From there, a variety of Web-based attacks were launched using an easy-to-purchase malicious toolkit, called the Eleonore Exploit Pack. Only first-time users were affected; returning to the site a second time did not lead to more attacks, making it difficult for law enforcement to track the perpetrators.

For less $1,000 – the Eleonore Exploit Pack costs only $700 – even the most minimally talented hacker can exploit flaws in Microsoft Internet Explorer, Firefox and Adobe Acrobat Reader, The widespread problem of low cost hacking that takes advantage of this commonly used software was highlighted in the 2010 Symantec report.

Despite the inherent risks involved in cloud platforms, IT experts tend to agree that the government would reap more benefits from using them, rather than not, and have encouraged government agencies to move towards the cloud in recent months.

"I am not going to say this will scare users away from cloud computing," says Thomas Krafft. "But it definitely brings into clear focus the issues

CA Security Advisor Research Blog
Get information on the latest threats

Zeus "in-the-cloud"

Published: December 09 2009, 04:39 AM
by Methusela Cebrian Ferrer

A new wave of a Zeus bot (Zbot) variant was spotted taking advantage of Amazon EC2's cloud-based services for its C&C (command and control) functionalities.

amazon web services™

This notable scheme is a highlight from the latest spammed executable "xmas2.exe" (63,488 bytes), for which we have recently published blog titled "Christmas is knocking on the door, so does the malware".

Evil greeting card arrives to users' mailbox

Entices users to click a malicious URL which links to a hacked legitimate website perpetrated for criminal activity such as serving Zeus bot variant.

Once executed, the Zeus bot variant will communicate to its C&C server, which in this case is controlled using "in-the-cloud" based services.

[Figure 01 – Zeus displays cyber-criminal activities]

Action	URL	Details
GET	http://ec2-[redacted]-170.compute-1.amazonaws.com/zeus/config.bin	svchost.exe (sr
POST	http://ec2-[redacted]-170.compute-1.amazonaws.com/zeus/gate.php	svchost.exe (sr
POST	http://ec2-[redacted]-170.compute-1.amazonaws.com/zeus/gate.php	svchost.exe (sr
POST	http://ec2-[redacted]-170.compute-1.amazonaws.com/zeus/gate.php	svchost.exe (sr
POST	http://ec2-[redacted]-170.compute-1.amazonaws.com/zeus/gate.php	svchost.exe (sr

[Figure 02 – Zeus bot variant communication]

As shown in Figure 03, the Zeus bot variant injects code into the system processes (such as svchost.exe) and connects to its cloud-server [Figure 02] for configuration (config.bin) of the master for it's criminal activity.

How could you data be leaked?

1. Data leakage from Devices

Name: Haibo
Salary: 100$
Creditcard: 8621 4579
Location: Emei Mountain

3. Outsider Attacks

Name: Haibo
Salary: 100$
Creditcard: 8621 4579
Location: Emei Mountain

Hoho, Haibo is a poor professor!!

4. Interference Attacks

Data Processing Apps

Operating Systems

Virtual Machines

2. Insider Attacks

VMM

Hardware Devices

Cloud

Butler Lampson: Retroactive Security?

Access control doesn't work

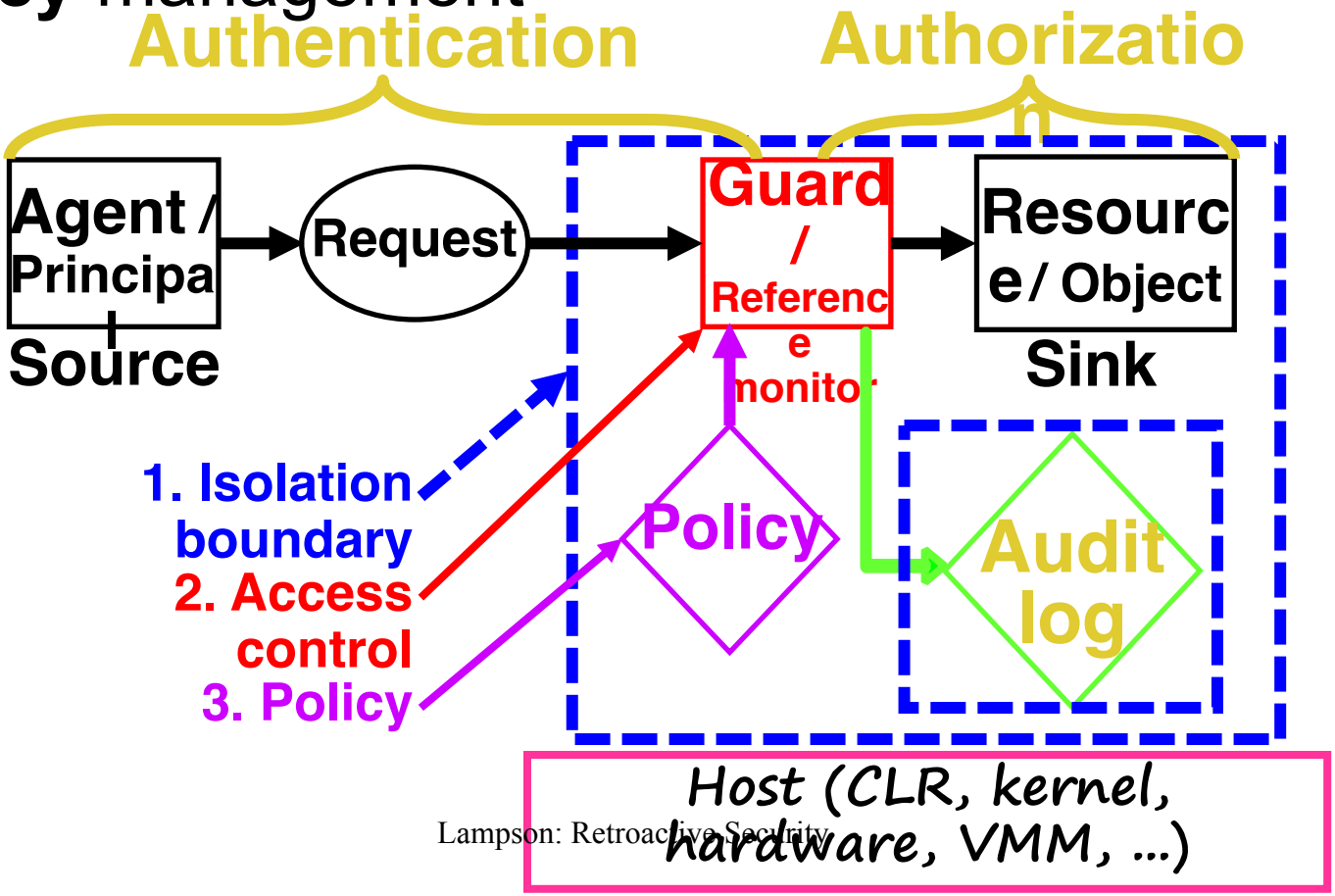
- 40 years of experience says so
- Basic problem: its job is to say "No"
  - This stops people from doing their work
  - and then they weaken the access control
  - usually too much, but no one notices
  - until there's a disaster

Retroactive security focuses on things that actually happened

- rather than all the many things that *might* happen

Butler Lampson: Access Control

1. **Isolation boundary** limits attacks to channels (no bugs)
2. **Access Control** for channel traffic
3. **Policy** management



Butler Lampson: Aspects of Retroactive Security

What about enforcing rules? Blame and punishment

- Assigning blame? Auditing
- Imposing punishment? Accountability

What about integrity? Selective undo

What about secrecy? Undo publication

What about bugs? Accountability and isolation

What about freedom? Red/Green

Issue#3: Power Efficiency

Power Efficiency

Greener World, Better Life!

Again, why it has to do with OS?

- OS control every computer and many devices!
- Thus determines the power used by each computer

Google Datacenter: Finland

Ocean Datacenter

Question: what's the energy consumption of per google search?

Light a 11W lighter for 15 minutes to 1 hour

Power Breakdown in Server

Component	Peak Power	Count	Total
CPU [16]	40 W	2	80 W
Memory [18]	9 W	4	36 W
Disk [24]	12 W	1	12 W
PCI slots [22]	25 W	2	50 W
Motherboard	25W	1	25 W
Fan	10 W	1	10 W
System Total			213 W

Table 1: Component peak power breakdown for a typical server

Sutardja-Dai Hall
UC Berkeley
93,000 sq. ft.
With Digital Controls

73% of US electricity is consumed in buildings

2/3 of buildings occupants complain

>70% of large buildings have digital controls

12 Variable Speed Fans

138 Air Dampers

312 Light Relays

50 Electrical Sub-meters

151 Temperature Sensors

> 6,000 Sense and Control Points

Why Operating System Matters?

Dynamic voltage and frequency scaling

- Adjust the frequency of CPU according to workload

Power off devices to save power

- Like disk, LCD, or even memory

Problem: energy consumption doesn't scale with workloads

- Static power, leakage power

New!: Near Threshold Voltage

Issue#4: Mobility

Smartphone OS

Symbian

Windows Mobile

RIM Blackberry OS

Apple iOS

Google Android

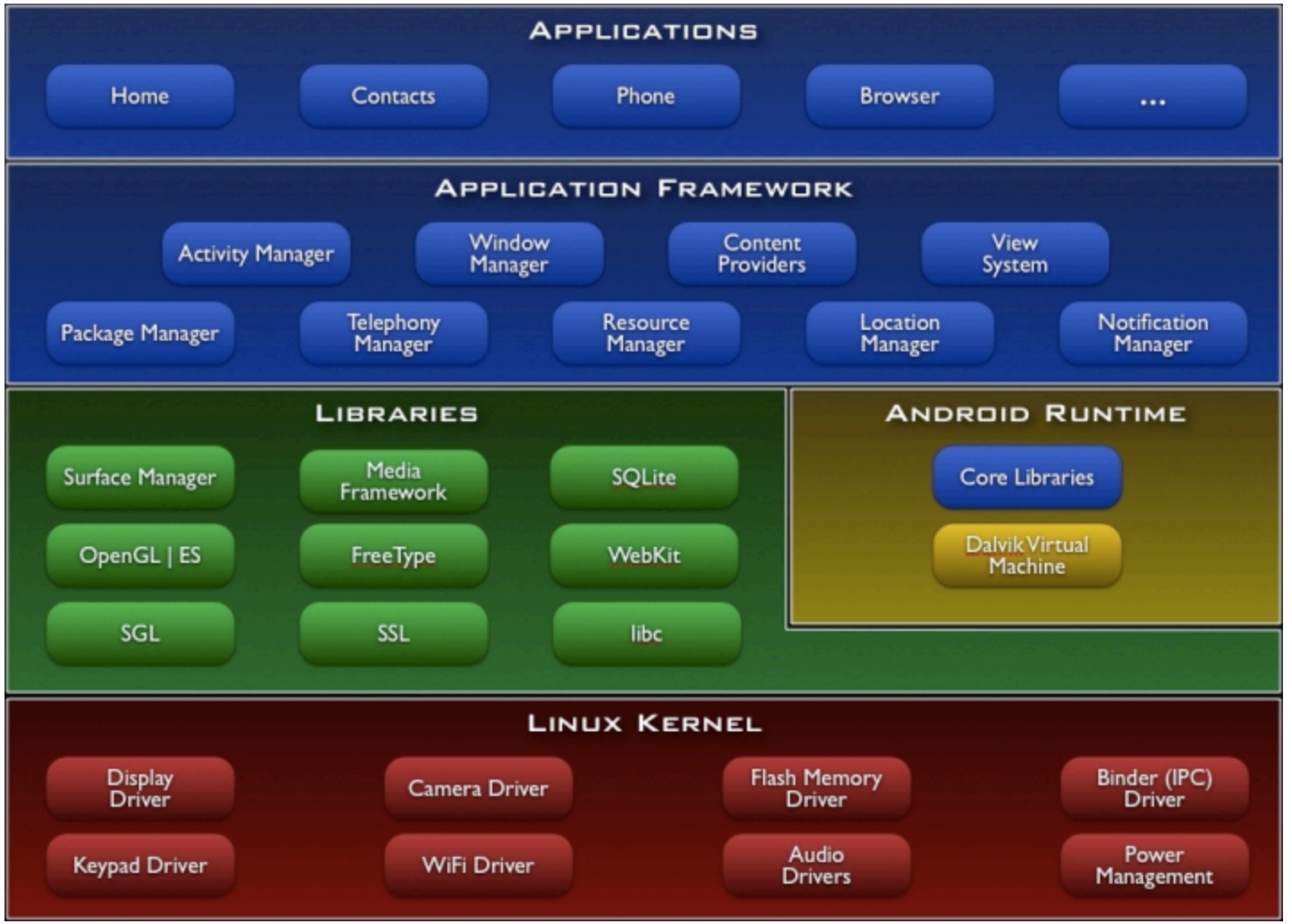
Palm WebOS

Windows Phone 7

Android Software Stack

- Linux kernel
- Libraries
- Android run time
  - core libraries
  - Dalvik virtual machine
- application layer
- application protocol

Android Architecture



What's Unique for Mobile OS?

Energy Efficiency

Richer User Experience

- But more limited resources

Security

- More user data put into mobile OSes

Issue#5: Write Correct Parallel Code

Writing concurrent program is hard

Concurrent programs are prone to concurrency bugs

Concurrency bugs have notorious characteristics

- Non-deterministic
- Hard to test and diagnose

But in multi-core era, we need to write concurrent program to harness the power of multi-core

Quote from Dijkstra

I am convinced more than ever that this type of work is very difficult, and that every effort to do it with other than the best people is doomed to either failure or moderate success at enormous cost.

Edsger Dijkstra
The Structure of the "THE" –Multiprogramming System
1968

Easiest One

Thread 1	Thread 2

`if (thd->proc_info) {`
`…`
`fputs(thd->proc_info, …)`
`…`
`}`
	`thd->proc_info = NULL;`

MySQL ha_innodb.hpp

Approaches to addressing the problem

- Concurrency bug detection
  - Race detection
  - Atomicity violation detection
  - Deadlock bug detection
- Concurrent program testing
  - Exhaustive testing
  - Different coverage criteria proposed
- Concurrent programming language/model design
  - Transactional memory

What OS can do?

Reduce sources of non-determinism

Tolerate application bugs

Faithfully capture concurrency bugs for reproduce

Provide better programming interfaces to ease programming

- Operating transactions

Issue#6: Scale Out (use distributed systems)

The Datacenter as a Computer

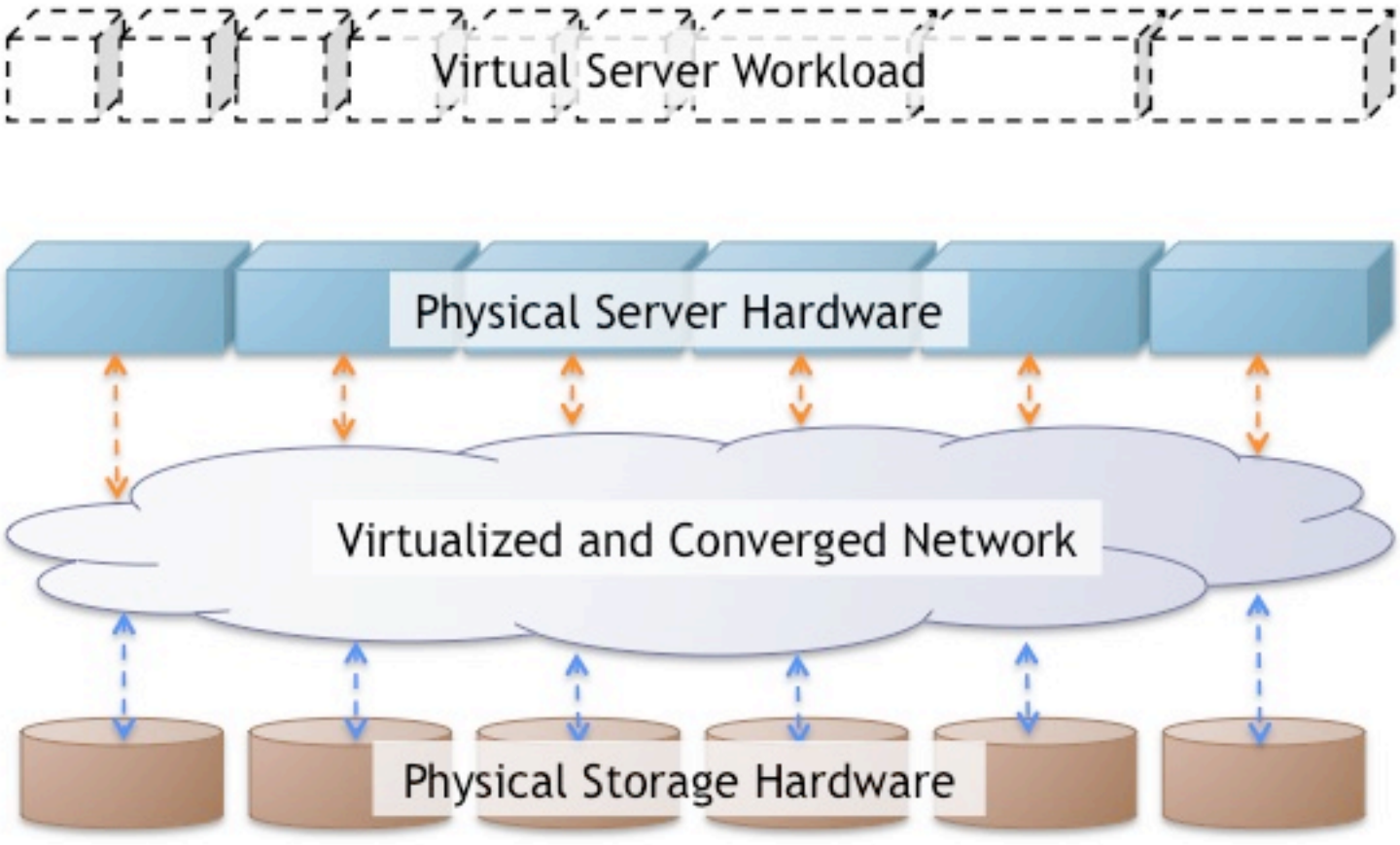
An Introduction to the Design of Warehouse-Scale Machines

Luiz André Barroso and Urs Hölzle
Google Inc.

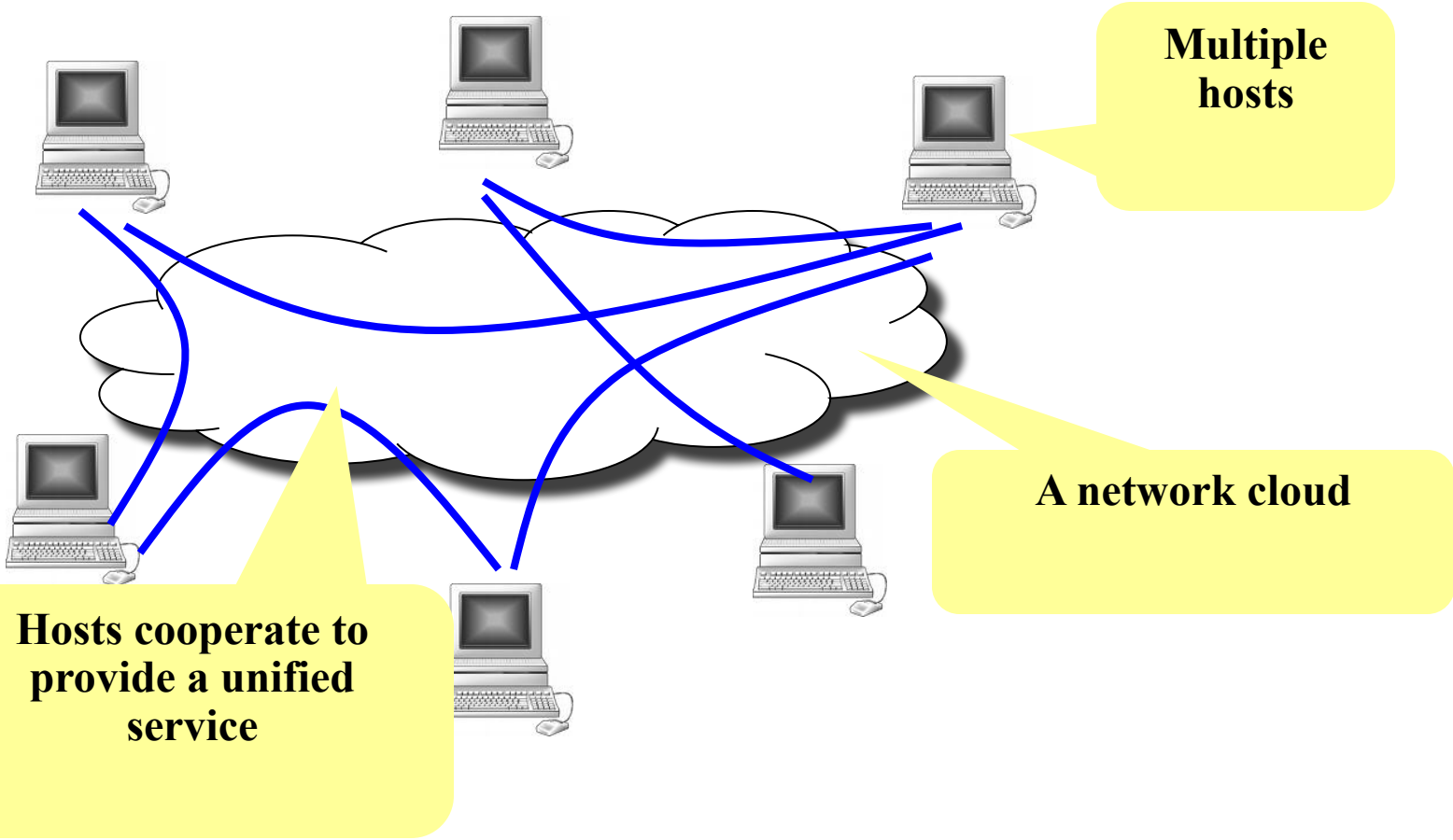
SYNTHESIS LECTURES ON COMPUTER ARCHITECTURE # 6

MORGAN & CLAYPOOL PUBLISHERS

The Virtual Datacenter Dream



What are distributed systems?



- Examples?

Why distributed systems? for ease-of-use

Handle geographic separation

Provide users (or applications) with location transparency:

- Web: access information with a few “clicks”
- Network file system: access files on remote servers as if they are on a local disk, share files among multiple computers

Why distributed systems? for availability

Build a reliable system out of unreliable parts

- Hardware can fail: power outage, disk failures, memory corruption, network switch failures…
- Software can fail: bugs, mis-configuration, upgrade …
- To achieve 99.999% availability, replicate data/computation on many hosts with automatic failover

Why distributed systems? for scalable capacity

Aggregate resources of many computers

- CPU: Dryad, MapReduce, Grid computing
- Bandwidth: Akamai CDN, BitTorrent
- Disk: Frangipani, Google file system

Why distributed systems? for modular functionality

Only need to build a service to accomplish a single task well.

- Authentication server
- Backup server

Challenges

System design

- What is the right interface or abstraction?
- How to partition functions for scalability?

Consistency

- How to share data consistently among multiple readers/writers?

Fault Tolerance

- How to keep system available despite node or network failures?

Challenges (continued)

- Different deployment scenarios
  - Clusters
  - Wide area distribution
  - Sensor networks
- Security
  - How to authenticate clients or servers?
  - How to defend against or audit misbehaving servers?
- Implementation
  - How to maximize concurrency?
  - What's the bottleneck?
  - How to reduce load on the bottleneck resource?

A word of warning

A distributed system is a system in which I can't do my work because some computer that I've never even heard of has failed."

-- Leslie Lamport

Trends on Distributed Systems

Convergence between traditional distributed systems, database systems, systems

- NoSQL: consistency vs. database
- Scale-out: MapReduce/Dryad
- PL with DS: translating high-level languages to MR

Goal: managing planet-scale information

- So called “big data”

Goal: delivering information timely

- Mobile Internet
- Mobile cloud: assisting mobile with cloud

Issues with Distributed Systems

Scalability

Fault-Tolerance

Geographic Distribution

Performance

Consistency

Issue#7: Non-Volatile Storage

Tape is Dead
Disk is Tape
Flash is Disk
RAM Locality is King

Jim Gray
Microsoft
December 2006

In memory of Jim Gray

Tape Is Dead
Disk is Tape

1TB disks are available

10+ TB disks are predicted in 5 years

Unit disk cost: ~$400 → ~$80

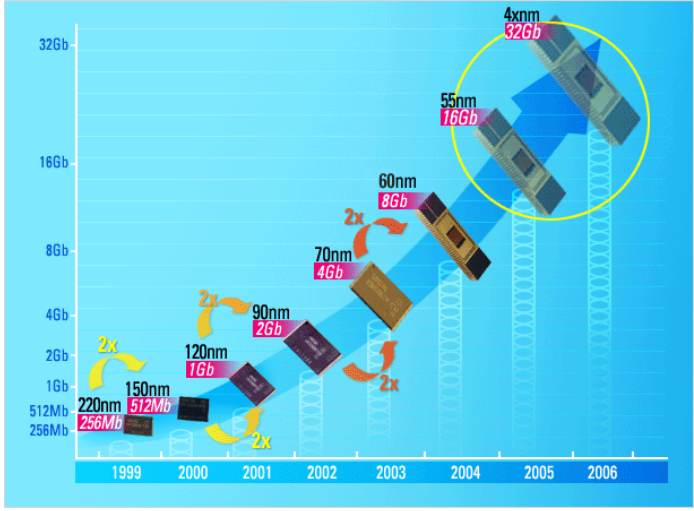
But: ~ 5..15 **hours to read (sequential)**

~15..150 **days to read (random)**

Need to treat **most of disk** as

Cold-storage archive

FLASH Storage?



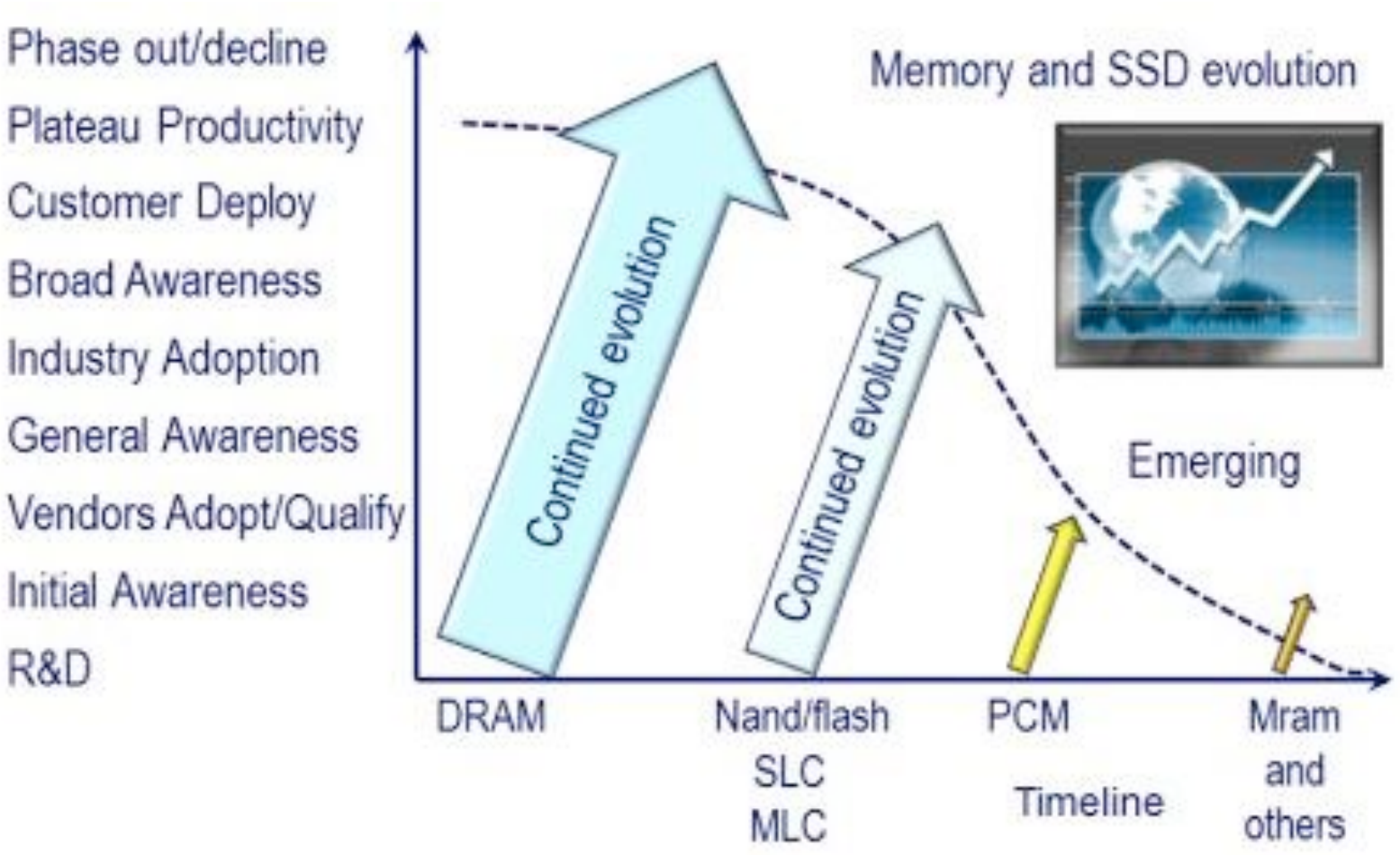
1995 16 Mb NAND flash chips
2005 16 Gb NAND flash
Doubled each year since 1995

Market driven by Phones, Cameras, iPod,…
Low entry-cost,
~$30/chip → ~$3/chip

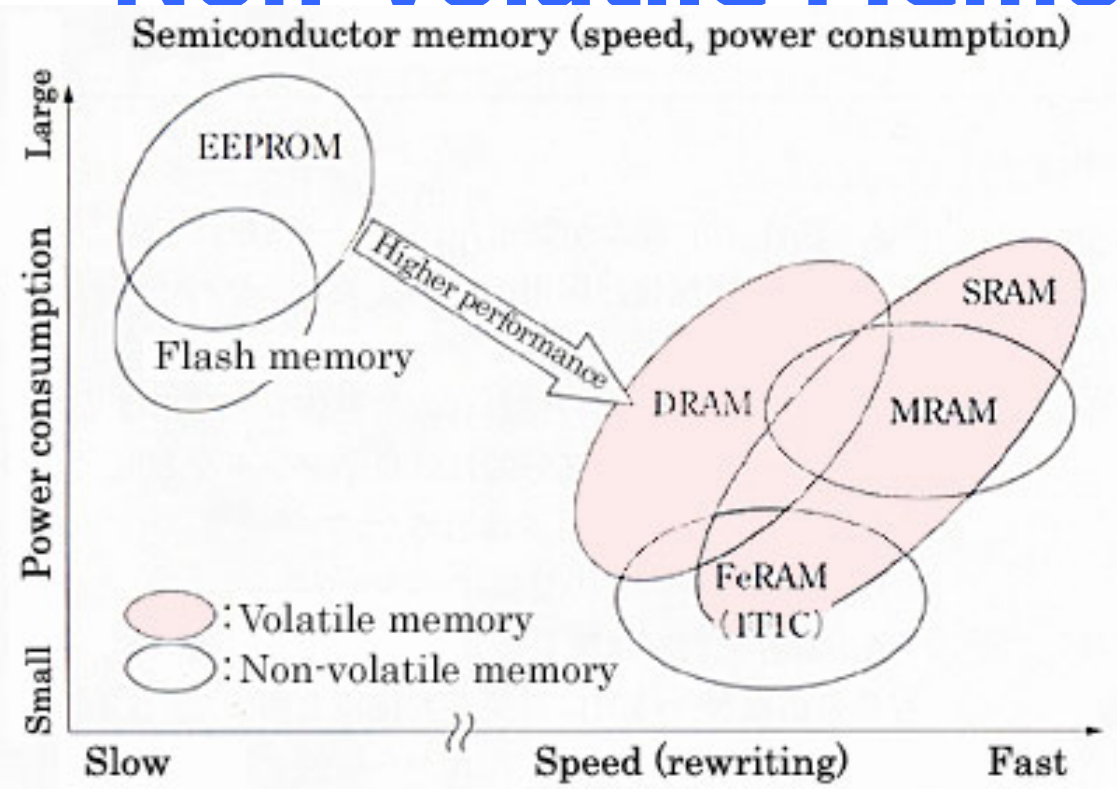
2012 1 Tb NAND flash
== 128 GB chip
== 1TB or 2TB “disk”
for ~$400
or 128GB disk for $40
or 32GB disk for $5

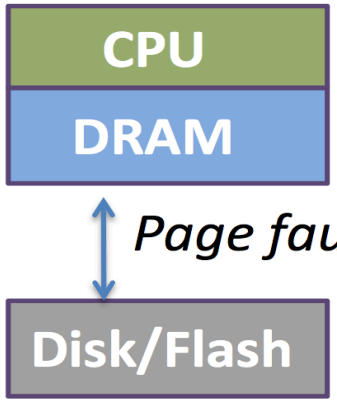
Samsung prediction

Storage Hierarchy



Non-volatile Memory





Memory

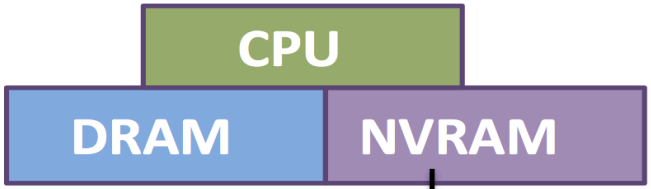
Load/store

Not persistent

Storage

Fopen, fread, fwrite, …

Persistent



STT-MRAM, PCM, ReRAM, etc.

Persistent memory

Load/store

Persistent

What OS can do?

Needs to adapt to the new storage model

Non-volatility changes the way operating system manage resources

- Like crash recovery
- Recall fsck

Better I/O performance

Issue#8: Virtualization

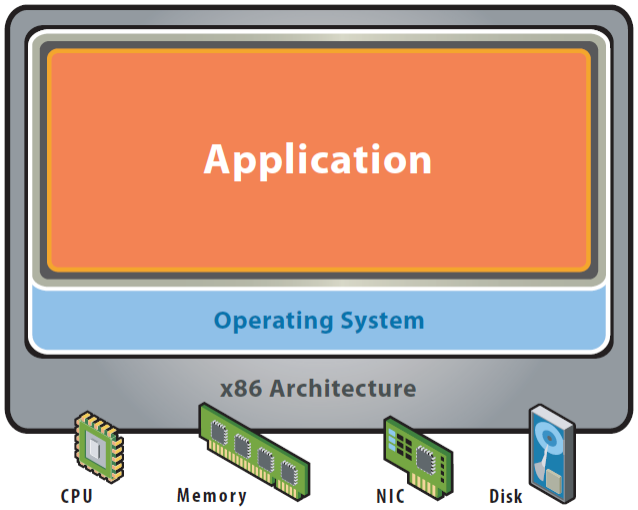
”All problems in computer science can be solved with another level of indirection.”

– *David Wheeler in Butler Lampson’s 1992 ACM Turing Award speech.*

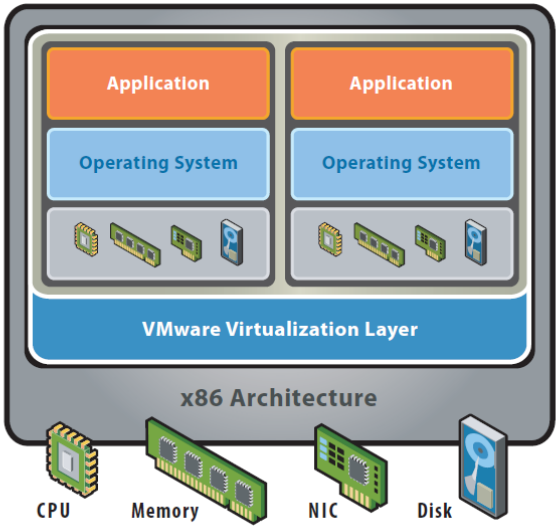
Concept of Virtualization

"Virtualization" refers to the creation of a virtual machine

A virtual machine is a software implementation of a machine (i.e. a computer) that executes programs like a physical machine



Before Virtualization



After Virtualization

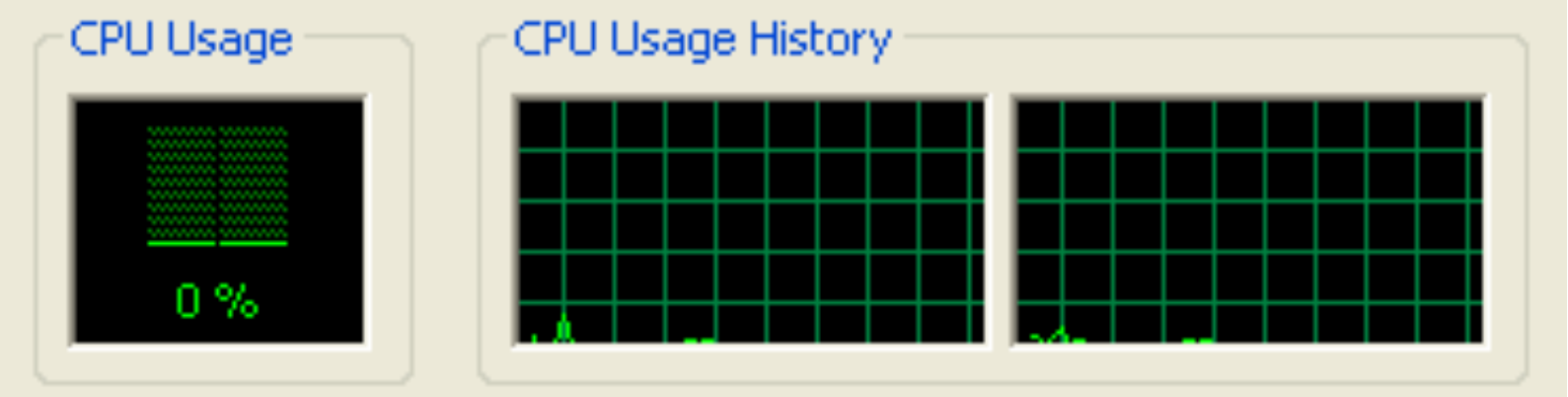
* "Virtualization in education", IBM, 2007
•figures from: VMware white paper, virtualization overview

Why Virtualization ?

- Decoupling the operating systems from physical hardware supports
  - Maximum resource utilization
    - Multiple OS environments can co-exist on the same computer, in strong isolation from each other
    - Server consolidation
  - Fault tolerance
    - Fault and migrate
  - Portability
    - Can use any OS
  - Manageability
    - Easy to maintain

Why Virtualize?

Too many servers for too little work



High costs and infrastructure needs

- Maintenance
- Networking
- Floor space
- Cooling
- Power
- Disaster Recovery

Virtualization is Essentially Another Layer of OS

How does the process and OS use hardware resources?

	process	OS
CPU	Non-privileged registers and instructions	+Privileged registers and instructions
memory	Virtual memory	+Physical memory
exceptions	Signals, errors	+Traps, interrupts
I/O	File system	Programmed I/O, DMA, interrupts

Next Class

File Systems

- Read paper FlashVM