Video Games – From Spacewar! to Today
The Japanese Invasion - Shigeru Miyamoto
id Software, Valve, and the Future
The Turing Test: A Timely Study

Adi Fuchs

COS583 Great Moments in Computing, Spring 2015
The Test Itself

- How does it reflect the progress in AI research?

- Does it say anything about consciousness?

- Does it say nothing at all, aside from human gullibility?
Passing the Turing Test

- Will it mark the rise of the machines?
- Is it even possible?
- How would we know if we did?
The Loebner Prize

- The annual “Imitation Game” Olympics, first held in 1991

- X terminals, at least 2 are human and 2 are computers

- Several judges, each talks to the terminal

- This year, 33% of the judges mistaken a machine to be human
Citation Chain

- Each cites its predecessor and the original paper
- 1950: “Can machines think?”
- 1991: “Let’s create a ‘practical Turing test’”
- 1992: “Practicing the Turing test makes no sense”
- 2000: “Maybe we should change the Turing test?”
- 2012: “It’s time for the Turing test to go...”
Cryptography -> Bitcoin

Ben Stallworth
Bitcoin

- Distributed Electronic Currency
- Public Ledger of Transactions
- Peer to Peer System
- Currently Supporting Burgeoning Start-up Industry
How does Bitcoin relate to a great moment?

• Application of ideas/principles from Diffie-Hellman
  • Distributed System of Trust
  • Verifying Identity using Public/Private Keys
  • Hash and One-Way Functions
  • Secure Digital Signatures

• Ideas Developed
  • Implementation of Digital Signatures
  • Hash Functions from One-Way Functions
  • Time Stamping and Identity Verification
Papers (in Citation Order)

- Diffie-Hellman. “New directions in cryptography”
- Merkle. “Secrecy, Authentication, and Public Key Systems”
- Goldwasser, Micali, et al. “A Secure Digital Signature Scheme”
- Rompel. “OWF are Necessary and Sufficient for Secure Signatures”
- Haber, Stornetta. “How to Time-Stamp a Digital Document”
- Haber, Stornetta. “Secure Names for BitStrings”
The Internet to Distributed Data

How Locating Objects in a Network Has Evolved

Cody Wilson
COS 583
Prof. Martonosi
April 27th, 2015
Step 1: Protocols

  - Router Solicitations and Advertisements
  - ICMP on top of IP
- RFC 1256 → Locating nearby copies of replicated Internet servers[3]
  - Anycast
  - Routing Probes - traceroute
  - Triangulated distance metrics
Step 2: Data Structures

- Nearby Copies of Servers $\rightarrow$ Nearby Copies of Replicated Objects[4]
  - From a central server of distances to tree of objects on each node

- Nearby Copies of Replicated Objects $\rightarrow$ Chord Distributed Hash Table[6]
  - From a tree of objects on each node, to a tree of nodes to find objects
  - Scalability
References

A Historical Tour of RSA Vulnerabilities

James Evans & Charles Marsh
COS 583
Spring 2015
Blind signatures (Chaum)
RSA (Simmons) 1978

Common modulus (Simmons) 1978

Blind signatures (Chaum) 1983
RSA
1978

Common modulus
(Simmons)
1983

Wiener’s attack
(Wiener)
1990

Blind signatures
(Chaum)
Capture the Flag (CTF)

The Backstory

On January 3rd, 1978, a large bank lost $100 million when a hacker broke into its database. Based on a tip, the FBI has investigated a cryptography researcher by the name of Bob Badguy for the crime, but they haven't managed to find much evidence in the 37 years since. Badguy is known to have had a keen interest in the emerging field of public-key cryptography, and covered his tracks using the cutting-edge cryptography available every step of the way for the past four decades.
Capture the Flag (CTF)

1. Deterministic attack: find Badguy’s password

2. Common modulus attack: decrypt identical messages to accomplices

3. Iterated encryption: decrypt his wire transfer metadata

4. …
Capture the Flag (CTF)

1. Deterministic attack: find Badguy’s password

2. Common modulus attack: decrypt identical messages to accomplices

3. Iterated encryption: decrypt his wire transfer metadata

4. …
Goals

• Learn about the academic history of RSA
• Learn about vulnerabilities in the protocol
• Create an educational tool for others

Welcome to Bobby B's server!

Username: bbadguy
Password: (plaintext password)
Submit
Translation in the Face of Varying Memory Consistency Models

Caroline Trippel
Project Status Lightening Round
04.27.2015
The first MCM: Sequential Consistency

MCMs specify the allowed behavior of a multithreaded program executing with shared memory [Sorin et al., 2011].

First defined by [Lamport 1979], execution is the same as if:

(R1) Memory operations of each individual processor appear in program order (PO)
(R2) Memory operations of all the processors were executed in some global sequential order

<table>
<thead>
<tr>
<th>Program</th>
<th>Legal Executions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thread 0</td>
<td>Thread 1</td>
</tr>
<tr>
<td>x=1</td>
<td>x=1</td>
</tr>
<tr>
<td>r1=y</td>
<td>r1=y</td>
</tr>
<tr>
<td>y=1</td>
<td>r1=y</td>
</tr>
<tr>
<td>r2=x</td>
<td>r2=x</td>
</tr>
<tr>
<td>y=1</td>
<td>x=1</td>
</tr>
<tr>
<td>r1=y</td>
<td>r2=x</td>
</tr>
<tr>
<td>r2=x</td>
<td>y=1</td>
</tr>
<tr>
<td>r2=x</td>
<td>y=1</td>
</tr>
<tr>
<td>r2=x</td>
<td>r2=x</td>
</tr>
</tbody>
</table>
Memory Consistency Models

C11/C++11 | Java | Cuda | OpenCL | ...  
---|---|---|---|---
LLVM IR | JVM | PTX | SPIR | ...

x86 CPU | ARM CPU | Power CPU | Nvidia GPU | AMD GPU | ...

Shared Virtual Memory
Memory Consistency Models

- C11/C++11
- Java
- Cuda
- OpenCL
- ...
- LLVM IR
- JVM
- PTX
- SPIR
- ...
- x86 CPU
- ARM CPU
- Power CPU
- Nvidia GPU
- AMD GPU
- ...

Shared Virtual Memory
Memory Consistency Models

- C11/C++11
- Java
- Cuda
- OpenCL
- ...

- LLVM IR
- JVM
- PTX
- SPIR
- ...

- x86 CPU
- ARM CPU
- Power CPU
- Nvidia GPU
- AMD GPU
- ...

Hardware MCMs

Shared Virtual Memory
Memory Consistency Models

Navigation of this space:
• Already difficult
• Worsened by increasing heterogeneity
1. The ArMOR syntax [Lustig et al., 2015]
   - Framework for specifying, comparing, and translating between memory consistency models.
   - Architecture-independent yet precise format for specifying the semantics of memory ordering requirements (MORs).
2. Dynamic translation with ArMOR [Lustig et al., 2015]
3. Install-time static translation with ArMOR
ArMOR ISA-Assisted Optimizations

- What if we had access to compiler metadata?
- **Thread private accesses**
  - Relatively straightforward compiler analysis can classify as many as 81% of memory accesses [Singh et al., 2012] as thread-private
  - We found that an average of 75% of memory accesses were stack accesses
- **Data-race-free code**
  - C++ execution guaranteed to be SC for non-DRF programs, else no semantics are given [Boehm and Adve, 2008]
  - Parsec benchmark suite written in C++
SKETCHPAD
IN JAVA

David Fridovich-Keil
ELE 583, Princeton University
No arcs, but...

• Point, line, composite shapes
• Point and line constraints
• Graph-based constraint satisfiability tester
• General, home-made nonlinear optimization for constraint satisfaction
Demo

http://youtu.be/GPbVSHIHLAY
Spacewar!

Dan Kang
Zap! With a beautiful flash and appropriate noise, Jimmy’s spaceship disintegrated; Beth had won Spacewar again. The nine-year-olds were lying on the grass of a park near their home, their DynaBooks hooked together to allow each of them a viewscreen into the space world where Beth's ship was now floating triumphantly alone.
Zap! With a beautiful flash and appropriate noise, Jimmy’s spaceship disintegrated; Beth had won Spacewar again. The nine-year-olds were lying on the grass of a park near their home, their DynaBooks hooked together to allow each of them a viewscreen into the space world where Beth's ship was now floating triumphantly alone.
Zap! With a beautiful flash and appropriate noise, Jimmy’s spaceship disintegrated; Beth had won Spacewar again. The nine-year-olds were lying on the grass of a park near their home, their iPhones hooked together to allow each of them a viewscreen into the space world where Beth's ship was now floating triumphantly alone.
Current Progress

- The ship can turn!
- The ship can thrust!
- The ship can fire!
- The ship is lonely...
Future Work

- Multiplayer!
- A star!
- Collisions and explosions!
DEC PDP-10: KL-10 (DECSYSTEM-20)
Conway’s Game of Life

- A dead cell with exactly three live neighbors becomes a live cell (birth).
- A live cell with two or three live neighbors stays alive (survival).
- In all other cases, a cell dies or remains dead (overcrowding or loneliness).

[Sources: math.com/students/wonders/life/life.html]
Virtual Memory

- 1961: Kilburn/Atlas
- 1967: MULTICS
- 1977: VAX minicomputer
So far
TODO

- Improve animation
- PDP-10
Citation Chain from Lamport’s Sequential Consistency Paper

Yatin Manerkar
Citation Chain (All Links)

- Lamport79
- Dubois86
- Goodman89
- Gharachorloo90
- Adve90
- Manson05
- Gharachorloo92
- Boehm08
Work Completed and Work Remaining

• Have read through papers to understand major steps forward at each point in the chain
• Need to re-read to understand the smaller and more subtle improvements between papers
Cryptography & Information Theory

Tess Marchant
Alice
+ Common paint
= Secret colours

Bob
+ Common paint
= Secret colours

Public transport
(assume that mixture separation is expensive)

Common secret
“[Communication theory and cryptography] were so close you could not separate them.”
– Claude Shannon
• Introduction
  – Importance of crypto alongside all information theory

• Communication Theory of Secrecy Systems
  – Shannon’s one-time pads
  – Reference intro

• Protocols for public-key cryptosystems & New Directions in Cryptography
  – Explain need for public-key cryptosystems/infeasibility of one-time pads in the modern world
  – Use paint metaphor to explain public-key system
  – Relate to information theory/scaling

• A Method for Obtaining Digital Signatures and Public-Key Cryptosystems & How to share a secret
  – Relate to MDH through the need for authentication alongside encryption
  – Explain the RSA algorithm
  – Relate to intro through need to trust people/entities that you share information with

• Quantum Cryptography
  – Describe findings in the paper
  – Explain how Quantum could disrupt RSA, and what can be done to counteract it

• Conclusion
  – Reiterate development of crypto
  – Reiterate relationship between IT and crypto
MapReduce’s Parallel Database Origins

Nevin Li
Background

• MapReduce – Programming model and implementation for processing large data sets in a parallel and distributed fashion on a cluster.
  • Map inputs into a list of pairs grouped by key, Reduce all pairs with the same key in some fashion.
• Meant to be read in conjunction with PageRank, but cut due to time constraints.
• David DeWitt blog post criticizing MapReduce for not being novel and using techniques the distributed database community have been using for a long time.
Paper Breakdown (Part 1)

• Application of Hash To Database Machine ... (1983)
  • Hashing tuples by key into same buckets to optimize joins.

• Multiprocessor Hash-Based Join Algorithms (1985)
  • Optimizations made to such hash algorithms.

• The Case for Shared Nothing Architecture (1986)
  • Multiprocessor systems should not share memory or disk.
Paper Breakdown (Part 2)

• Prototyping Bubba, A Highly Parallel Database System (1990)
  • Early parallel database prototype featuring the hash-based join algorithms.
• High-Performance Sorting on Networks of Workstations (1997)
  • MapReduce-like backend involving source workers partitioning the data to be sorted and sent to reduce workers.
Teleportation Operations

Source

$|q_1\rangle$

$H$

$|q_2\rangle$

$Z$

$|q_3\rangle$

$X$

Destination

Graphic courtesy of Ali JavadiAbhari

$K = 4$

$D = 20$

Entangled EPR

Teleportation Source

Teleportation Unit

Global Memory

Ballistic Channel

G1

G2

G3

G4
Scheduling Problem

• To move qubits, need entangled pairs ready at source and destination
• Could move all entangled pairs at start
• Could deliver entangled pairs “on demand”
• My project: find good tradeoff
Progress To Date

• Selecting new project
• Reading background papers
• Compiling code
  – New dev account
• Understanding code
• Playing around with priority queue approach
Progress To Date

• Selecting new project
• Reading background papers
• Compiling code
  – New dev account
• Understanding code
• Playing around with priority queue approach

Next...

• Write a better smoothing algorithm
• Get results!
“Markerwitz”
by Dale Markowitz and Shubhro Saha

http://imgur.com/RJbB8Xs
Problems in Original Boolean Algebra

- $x+y$ is well defined only if $x$ and $y$ are disjoint. ✗
- $x-y$ is well defined only if $x$ includes $y$ as a subclass. ✗
- $x^2=x$ is the Fundamental Law of Thought but $x^3=x$ is meaningless! ✗
Some of my findings:

- **Logical Algebra is introduced by Peirce (1867)**
  - Logical addition: $x+y$ is defined even when $a$ and $b$ are not mutually exclusive ✓
  - Logical multiplication: $x.x.x.x.x.x.x.x=x$ ✓
  - Logical subtraction: $x-y$ is still uninterpretable if $y$ is not a subclass of $x$ ❌
  - Logical division: Trying to keep duality of operations ❌

- **J. Venn introduces Venn Diagrams (1881)**
  - Boole did not have Venn diagrams !!!
  - Venn still cannot not resolve $x-y$ issue. ❌
Some of my findings:

- Principia Mathematica re-defines symbolic logic (1912)
  - «OR» and «AND» are analogous to logical addition and logical multiplication.
  - Other logical operations are: «NOT», «IMPLIES» and «EQUIVALENCE».
  - There is no counterpart of logical subtraction or logical division.
To do:

- Project is mostly finished, spell check is required!
- If there are interested people, I can send them a copy of the latest draft.
Thank you for listening!
References:


Enigma Simulator

COS 583
Thee Chanyaswad
What is the Enigma?

• An electro-mechanical enciphering/deciphering device used by the German during WWII.

Why is it important?

- The Enigma prompted Alan Turing and teams of code-breakers to develop machines to break it.
- Many of the team members went on to build the world’s first generation of computers [1].

The Enigma Basics

- **Input keyboard**
- **Stecker board**
- **Reflector**
- **Entry disc**

Signal flow:

- 1 Input keyboard
- 2 Stecker board
- 3 Entry disc
- 4 Reflector
- 5 Stecker board
- 6 Output lamps
The Simulator

• Written in C++
• Done so far...
  ▪ Created a class for each of the Enigma parts.
  ▪ Finished the main operation function.
  ▪ Wrote the majority of the display( ) function.
• To be finished...
  ▪ The display( ) function.
THANK YOU.
Lightning Round

Themistoklis Melissaris
Project outline

• “From Weiser’s dream to the Cloud”
• Evolution of the field of ubiquitous computing
• Effects on ubiquitous computing by:
  – Hardware
  – Services
  – New concepts
Papers

• Some Computer Science Issues in Ubiquitous computing, 1993
• Context-Aware Computing Applications, 1994
• People, Places, Things: Web Presence for the Real World, 2000
• The Personal Server: Changing the Way We Think about Ubiquitous Computing, 2002
• Calling the Cloud: Enabling Mobile Phones as Interfaces to Cloud Applications, 2009
Hardware/Concepts Timeline

Xerox PARC prototypes
PARCtab

Birth of UbiComp
Context-aware computing
Web Presence
Personal server
Computation offloading
To Do

• Are there any Weiser influences?
• Any predictions fulfilled?
• Citation chains and relationships
• My predictions
RAID Teaching Tool

By Utsarga Sikder
RAID LEVEL 0

Disk 0

Log

Read

Write

Corrupt
Wrote “hello world” to disk 0, position 0.
Wrote data “hello world” to disk 0, position 0. Switched to RAID L1.
Wrote “hello world” to disk 0, position 0.
Switched to RAID L1.
Read data “hello world” from d0, p0.
Read redundant data “hello world” from d1, p1.
Switched to RAID L1.
Read data “hello world” from d0, p0.
Read redundant data “hello world” from d1, p1.
Corrupting disk 0. Recovered data using disk 1.
Read data “hello world” from d0, p0.
Read redundant data “hello world” from d1, p1.
Corrupting disk 0. Recovered data using disk 1.
Dumping disk 1: “hello world”
Tracing of data models of database
Nosql movement

• Michael Stonebraker ‘s talk at EPFL
• data warehouses will migrate to column-based data stores within 10 years. The traditional row-based data storage approach is dead, as row-based storage will never match column-based storage’s performance increase by factor 100x.
• the race for the best data storage designs has not yet been decided, but there is a clear indication of classic models being “plain wrong” (according to Stonebraker), as only 4% of wall-clock time is spent on useful data processing, while the rest is occupied with buffer pools, locking, latching, recovery.
## Data Models Tracing

<table>
<thead>
<tr>
<th>Year</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1970</td>
<td>Relational</td>
<td>A Relational Model of Data for Large Shared Data Banks</td>
</tr>
<tr>
<td>1976</td>
<td>Entity-relation</td>
<td>The Entity-Relationship Model - Toward a Unified View of Data</td>
</tr>
<tr>
<td>1981</td>
<td>semantic</td>
<td>Database Description with SDM: A Semantic Database Model</td>
</tr>
<tr>
<td>1983</td>
<td>Extended Relational</td>
<td>The Database Language GEM</td>
</tr>
<tr>
<td>1986</td>
<td>Object Oriented</td>
<td>An object server for an object-oriented database system</td>
</tr>
<tr>
<td>1986</td>
<td>Object Relational</td>
<td>The Design of Postgres</td>
</tr>
<tr>
<td>1997</td>
<td>Semi Structured</td>
<td>A Database Management System for Semi-structured Data</td>
</tr>
<tr>
<td>2006</td>
<td>Column Oriented</td>
<td>Bigtable: A Distributed Storage System for Structured Data</td>
</tr>
</tbody>
</table>
1970 A Relational Model of Data for Large Shared Data Banks

• **Relational database**
• Store the data in a simple data structure
• high level set-at-a-time DML
• No need for a physical storage proposal
1976 The Entity-Relationship Model - Toward a Unified View of Data

- **Entity relation**
- a collection of instances of entities
- entities have attributes
- there could be relationships between entities (1-to-1, 1-to-n, n-to-1 or m-to-n)
- Relationships can also have attributes
1981 Database Description with SDM: A Semantic Database Model

• **Semantic**
• relational data model is “semantically impoverished”, incapable of easily expressing a class of data of interest.
• “post relational” data model - Semantic data models
• focuses on the notion of classes, which are a collection of records obeying the same schema
• Have aggregation and generalization
1983 The Database Language GEM

- **Extended relational**
- Adding a new “feature” to the relational model to correct the problem
- proposed adding the following constructs to the relational model, together with corresponding query language extensions
  - set-valued attributes
  - aggregation
1986 An object server for an object-oriented database system

• Object-oriented
• an “impedance mismatch” between relational data bases and languages like C++
• to bind an application to the data base required a conversion from “programming language speak” to “data base speak” and back
1986 The Design of Postgres

• Object relational
• Motivated by GIS queries, which are difficult to express in SQL
• the OR proposal added
  – user-defined data types,
  – user-defined operators,
  – user-defined functions,
  – and user-defined access methods
• First prototype was Postgres
1997 A Database Management System for Semi-structured Data

• Semi-structured
  – Schema later
  – graph-oriented