

# What Inspires Systems Research

Liviu Iftode

Rutgers University CS 545: Distributed Systems

# What is Systems Research

- n Anything that requires prototyping to proof
- n Computer Science or Engineering?
- n Good systems research: a combination of many tradeoffs
  - Address a real problem but do not abandon previously addressed problems
  - Novel but feasible and compatible with existing approaches
  - Multiple goals: performance is always important

## Systems Research Questions

- n What drives systems research towards success or failure?
- n Where do novel ideas come from and when?
- n What makes systems research results persist longer?
- n How to anticipate the next fashion in systems?

## Methodology

- n Take a 15-year retrospective view to the top OS conference, Symposium on Operating Systems Principles (SOSP)

## Eight SOSP Conferences

- n SOSP'91, Pacific Grove, CA
- n SOSP'93, Asheville, NC
- n SOSP'95, Copper Mountain, CO
- n SOSP'97, Saint-Malo, France
- n SOSP'99, Kiawah Island, SC
- n SOSP'01, Chateau Lake Louise, Canada
- n SOSP'03, Bolton Landing, NY
- n SOSP'05, Brighton, UK

## SOSP'91

- n Main topic: File systems
  - 7 out of papers 18 papers are related to file systems
  - [Log-Structured File System \[Rosenblum\]](#),
  - [Semantic File Systems \[Gifford\]](#)
  - [Disconnected Operation in Coda \[Satya\]](#)
- n On the rise
  - Multi-threading: [Scheduler Activations \[Anderson\]](#)
  - Multiprocessors: Software DSM [Munin \[Carter\]](#)
  - Real time: Multimedia
- n New problems:
  - Networking: Automated Reconfiguration [Schroeder]
  - Security: Authentication [Lampson]
  - Reliability: Replication in FS [Liskov]

# Log-Structured File System

- n Problem:
  - More memory -> larger caches -> **disk traffic dominated by writes**
  - Writes are synchronous (caches do not absorb writes) and dominated by small accesses (metadata) -> **slow**
- n Solution:
  - Buffer file changes together (data and metadata) and write them all sequentially in a single large disk write (log)
- n Advantages:
  - **Fast** (practically eliminates the seek operation)
  - Simple recovery
- n Secondary Issues
  - How to locate file blocks
  - How to manage the free space: garbage collection
- n Generated heated debates for several years (Seltzer vs. Ousterhout, 1993-1995)

# LFS: An Example of Good Systems Research

- n Driven by awareness of technology trend
- n A radical depart from traditional approach
- n Maintains compatibility with traditional approach
- n Good model analysis
- n Solid implementation and evaluation

# Semantic File Systems

- n Problem:
  - Simplify user's life
  - Associative access: access files by content
  - Pioneering work in manageability
- n Solution
  - Virtual directory names interpreted as queries
  - Automatic extraction and indexing of file attributes
- n Advantages
  - No significant performance loss
  - Compatible with existing file systems protocols

# Coda

- n Problem:
  - Mobile users: mobile computing causes disconnections
  - FS should remain usable in the presence of communication failures
- n Basic Idea
  - Availability more important than consistency
  - Use caching for availability
- n Solution
  - Cache whole files in advance (best-effort hoarding)
  - Emulate server at client when disconnected
  - Reintegration upon reconnection: resolve conflicts
- n Advantage
  - Simple, feasible, usable
  - Addresses the common case (optimistic approach): no file sharing
- n Signs of success
  - Recognized problem raised by an emerging field: mobile computing
  - Coda still distributed after 15 years: why?

## Too Complex Solutions to Real Problems?

- n Scheduler Activations
  - Problem: Neither user-level nor kernel-level threads are the right abstraction
  - Solution: A new kernel interface for user-level threads
- n Munin: release-consistency software DSM
  - Problem: too much false sharing due to large coherence granularity
  - Solution: convince programmer to accept a relaxed consistency model in order to get acceptable performance

## SOSP'91 Summary

- n New directions triggered by
  - **New technology**: increasing memory capacity, mobility
  - **User's problems**: semantic file systems
- n Main application domain of the time (high-performance computing) triggers complex solutions (multithreading, software DSM)

# SOSP'93

- n No main topic
- n Still there
  - File systems: Zebra[Ousterhout]
  - Performance: anticipates new OS structures
- n Interesting problems
  - Make networking fast: Fbufs[Druschel]
  - Models for distributed systems: Limits of causality and total ordered communication[Cheriton]
  - Location information in ubiquitous computing [Spreitzer]
- n Follow-up
  - Authentication in Taos[Lampson]
- n No further analysis

# SOSP'95

- n Main topic: OS structure and performance
  - Exokernel[Kaashoek] and SPIN[Bershad]
  - Impact of Architecture on OS Performance [Rosenblum]
- n On the rise
  - Reliability: Hypervisor[Schneider], Hive [Chapin], Logged VM[Cheriton]
  - Mobility: Bayou[Terry]
- n Still of interest
  - File systems: xFS[Wang], Informed Prefetching and Caching [Gibson]
  - Global Memory[Feeley] and DSM CRL[Johnson]
- n Follow-ups
  - Weak connectivity[Satya]
- n New problems
  - User-level communication: U-Net [Eicken]
  - 64-bit address spaces

# New OS Structures and OS Performance

- n Problem
  - Oss have become too big, rigid and hard to manage
  - Performance does not increase with raw hardware performance
- n SOSP'95 debate
  - Micro-kernel is fine, just needs a good implementation [Liedtke]
  - Nano-kernels with application-controlled resource management: Exokernel[Kashoek]
  - Safe extensibility through downloadable modules written in a type-safe language: SPIN[Bershad]
- n What happened after
  - Complex solutions with hardly feasible assumptions
  - Influenced future OS design but not clear how much: Why?
  - OS designers had to face a new challenge (network-centric server applications) and forgot the OS structure debate

# Research on OS Reliability on the rise

- n Problems: OS often fail, how to make applications survive?
- n Papers anticipate VMM research, which exploded ten years later
- n Hypervisor-based fault-tolerance [Bressoud&Schneider]
  - Interpose a VM software layer between HW and OS
  - Log non-deterministic events in order to mirror state of a primary computer onto a backup
  - Continued a decade later with the ReVirt [Chen] but applied to intrusion analysis
- n Hive[Chapin, Rosenblum]
  - Fault-containment using a cellular OS
  - Better resource allocation
  - A decade later: VMware Inc

## OS Research for Cluster Computing

- n OS research influenced by cluster computing demands for performance and programmability
  - TCP/IP too heavy protocol for high-performance interconnects
  - Message-based programming is too difficult for large scale
- n User-level communication: U-Net
  - Simple and efficient but requires remote memory access hardware
  - In use today: Infiniband for network storage
- n Software DSM and global memory
  - Make it simpler: eliminate restrictions imposed to the parallel programming model but require new API
  - Still didn't make it: Why?

## Complexity in OS design

- n xFS[Wang]
  - A completely decentralized distributed file system
  - Wonderful engineering work
- n Bayou[Perry]
  - Eventual consistency using an anti-entropy protocol for update propagation
  - Automatic resolution of update conflicts
  - Better than Coda?

## SOSP'95 Summary

- n An internally-generated problem (OS became too complex) raised concerns about OS structure and performance
- n Research in OS reliability anticipates new challenges: fault isolation and containment
- n Cluster computing only half-way successful in OS research
- n Research in mobility and file systems end-up proposing unfeasible solutions
- n Follow-up on your own research may not be a bad idea

## SOSP'97

- n No major topic
- n On the rise
  - OS support for Internet services: BASE instead of ACID [Fox], Security in Java[Wallach]
  - Real-time scheduling for multimedia
  - Application Adaptation for Mobility: Odyssey[Noble]
- n Still around
  - Running commodity OSES on multiprocessors: Disco [Rosenblum]
  - Software DSM: Shasta[Scales], Cashmere[ Stets]
  - Distributed file systems: Frangipani[Thekkath]
  - OS performance profiling
- n New problems:
  - Decentralized information flow-control[Myers]
  - Dynamic data race detection: Eraser[Savage]
- n Follow-up
  - Scalability in Exokernel[Kaashoek],
  - Flexible update propagation in Bayou[Petersen]

## New Research Problems

- n Decentralized Information Flow Control
  - Privacy becomes a concern in client-server interaction, especially when code migration is allowed
  - How to share information among systems with mutual distrust and no central authority
  - Static analysis can help
- n Detection of data races in multithreaded programs
  - An important class of bugs caused by programming error to follow a locking discipline
  - Exposure depends on scheduling non-determinism
  - Notorious hard to detect, reproduce, locate and eliminate
  - Two approaches
    - n Use happens-before relationship[Lamport]
    - n Eraser: track lock sets
  - Races-freedom not enough for correctness: we need atomicity [see ASPLOS'06]
- n Both papers prevent programming errors to cause damages, hot field today

## The Internet changes the OS

- n What OS changes Internet services require
  - Problem
    - n Clusters preferred to large MPs: cost-effectiveness
    - n ACID (atomicity, consistency, isolation, durability) data semantics is hard to support and not always required
    - n Availability is more important
    - n ACID precludes many performance optimizations
  - Solution:
    - n Relax data semantics for better availability
    - n BASE (basically available, soft state, eventual consistency)
- n Internet Services: one of the application area with the most significant impact on OS research
  - Lazy Receive Processing, Scout-OS, LARD, IO-Lite, Resource Containers
  - Security

## SOSP'97 Summary

- n Internet service applications are about to become the next main application domain
- n High-performance computing makes its last strong appearance in OS research (two DSM papers!)
- n A slowly but steadily emerging field: software bugs (detection)

## SOSP'99

- n Main topic: Internet Services
  - Manageability&Availability & Scalability:Porcupine[Levy]
  - Negative result: Cooperative web proxy caching[Levy]
  - Distributed VM for networked computers[Sirer]
  - Soft-timers for network processing[Aron]
- n On the rise
  - Security: separating key management from FS security [Mazieres], Eros-Fast capability[Shapiro]
  - Networking: Intentional Naming System[Balakrishnan], Click[Morris]
- n Still there
  - User errors: When to forget in Elephant FS [Santry]
  - Real time OS issues
- n New
  - Energy: Adaptation for mobile applications[Flinn]
- n Follow-up
  - Resource management in Cellular Disco[Govil]

## New Research Problems

- n Email is important (see next slide)
- n Battery lifetime: adapt applications
- n User mistakes
  - Elephant: Let system decide when/what to delete
- n Interrupts are expensive for network servers
  - Use soft-timers
- n Security for global file systems
  - SFS: Self-certifying pathnames eliminate need for key management
- n Naming in dynamic and mobile networks
  - INS: route messages by names
- n Flexible network routers
  - Click: implement routers in software

## Why email

- n Mail is important
  - Real demand
- n Mail is hard
  - Write intensive
  - Low locality
- n Mail is easy
  - Well defined API
  - Large parallelism
  - Weak consistency

# SOSP'01

- n Main topic: Peer-to-Peer and Overlay Networks
  - P2P storage systems: PAST[Rowstron], CFS[Dabek]
  - Resilient Overlay Networks [Andersen]
- n On the rise
  - Software Bugs: Bugs as Inconsistent Behavior [Engler], OS Errors[Engler]
- n Still around
  - Internet Services: Continuous Consistency [Yu], Event-Driven SEDA [Welsh]
  - OS adaptation: Gray-Box[Arpaci-Dusseau]
  - File systems and networking: Low-bandwidth Network FS [Mazieres]
- n New problems:
  - Energy conservation in hosting centers [Chase]
  - Sensor Networks: Low Level Naming [Heidemann]
- n Follow-ups
  - Privacy among untrusted hosts: Secure Programming Partitioning[Myers]

# P2P hijacked the OS

- n Decentralized storage systems
  - Main properties: scalable, highly available
  - Use scalable routing and lookup substrates (Pastry, Chord)
  - What is the real problem they solve?
- n Resilient Overlay Networking (RON)
  - Problem: Internet routing problem
  - Solution: Application-layer overlay on top of IP
  - Advantage claim: more resilient than IP routing
  - What is the OS problem?
- n Sensor Networks another hijacker?

## Practical Intellectual Challenges

- n Concurrency programming debate:
  - Threads or Staged Event-Driven (Seda)?
  - Event-driven manages load better
  - Practical
- n How to transfer file over a low-bandwidth connection?
  - Exploit similarities between files and file versions
  - Avoid sending data blocks over the network if they are already in cache at server/client

## ...and Less Practical Intellectual Challenges

- n Continuous consistency
  - Replication for availability makes consistency hard to strictly maintain
  - Trade consistency for availability
  - Metric: max deviation from strong consistency on replica-basis
- n How to acquire OS internal state info and control without modifying the OS?
  - Gray-box idea: interpose Information&Control Layers between client and the OS to exploit knowledge of the algorithms used by the box
  - Successfully used for controlling file caching, disk layout, etc

# Software Bugs

- n **Problem**
  - How to determine the correctness rules when programmers do not specify them
- n **Solution**
  - infer rules as “Programmer beliefs” from static analysis
  - Cross-check them for contradiction
- n **Evaluation**
  - Hundreds bugs are found in Linux: better than manual
- n **Question**
  - Dynamic monitoring(Eraser) or static analysis?
- n **Another paper on OS errors**
  - Device drivers have error rate 3-7 times higher than the rest of the kernel (see next SOSP)

# New Problems

- n **Conserving Energy in Hosting Centers**
  - Problem: Energy becomes the driving resource management issue
  - Solution: Adaptation to load by dynamically resizing the active server set with a certain degradation of service
- n **Sensor Networks**
  - Low-level naming based on attributes relevant to the application and external to the network topology (like INS but not over IP)
  - In-network processing of data: directed diffusion
  - What is the OS problem?

# SOSP'03

- n Main topic: OS Robustness
  - Execute untrusted code: Model-Carrying Code[Sekar]
  - VMM: Xen[Barham], Untrusted on XOM[Lie], Terra[Boneh]
  - Handle bugs in OS drivers: Nooks[]
  - Race condition detection: RacerX[]
  - Backtracking intrusions [Chen]
- n Still around
  - File Systems: Google File System[Ghemawat]
- n Follow-up
  - Policies into Mechanisms using Infokernel[Arpaci-Dusseau]
  - Overlay networks and P2P

# New and old hardware inspires OS research

- n Problem
  - OS not trusted
- n Three solutions
  - Virtual Machine Monitors: Xen
    - n X86 requires paravirtualization (an old OS research fashion)
  - OS over XOM processor architecture
    - n HW trusted to execute tamper-resistant SW
  - OS over a trusted VMM: Terra
    - n Tamper-resistant HW partitioned in multiple isolated VMs
    - n Applications can cryptographically authenticate the software stack to remote parties: Attestation

## More research on software bugs

- n Nooks
  - Problem: faulty drivers
  - Solution
    - n Fault resistance not fault tolerance
    - n Isolate driver failures with lightweight protection domain to prevent kernel corruption
- n Backtracker
  - Problem: analyze intrusions is hard
  - Solution: VM to log events and objects in dependency graphs
- n Static detection of race conditions and deadlocks: RacerX

## SOSP'05

- n Main topic: OS Security and Robustness:
  - OS integrity without HW: Pionner[Perrig]
  - Intrusion Detection and Containment: Vulnerability-Predicates[Chen], Vigilante[Costa]
  - Software bugs: Asbestos[Morris], Rx[Zhou]
- n Still around
  - Declarative overlays
  - Byzantine fault tolerance
  - Semantics in File Systems: Connections[Ganger]
  - Race detection with adaptive tracking: RaceTrack[Yu]
- n Follow-up
  - IRON File Systems [Arpaci-Dusseau]

## "No hardware" inspires OS

- n "Intellectual" Problem:
  - Verify code integrity
  - No trusted hardware support
- n Solution
  - All-software based code attestation using integrity measurements
  - Expected time of checksum code execution
- n Assumption:
  - Client (dispatcher) knows the configuration of the untrusted hardware

## Software bugs: New Approaches

- n Labels&Events
  - Problem
    - n Current OS abstractions do not provide sufficient flexible isolation between different users
  - Solution
    - n OS support for information flow control
- n Rx: treating bugs as allergies
  - Problem
    - n How to survive software failures safely
  - Solution
    - n Rollback, modify environment and re-execute
  - Idea
    - n Bug exposure depends on execution environment

# A New OS Problem: Intrusion Detection

- n Detecting intrusions using vulnerabilities predicates
  - Problem
    - n Prevent software bugs to be exploited by the attacker until they are fixed
  - Solution
    - n Predicates to monitor intrusions triggering the vulnerability
    - n Use VMM: IntroVirt
- n End-to-end containment of internet worms
  - Problem
    - n Internet worms containment must be done automatically because they spread too fast
  - Solution
    - n Collaborative worm detection and containment
    - n Self-certifying alerts: proof of vulnerability since hosts do not trust each other
    - n Use SCA to generate filters to block infection

# Preliminary Conclusions

- n A favorite theme
  - Created by a new technology, a new application domain
  - Sometimes its importance grows slowly
  - Dominates 1-2 SOSP cycles
  - After that, proposed solutions become too complex and less influential
- n Several permanent themes
  - File systems
  - Real time
- n OS research hijacking?
  - Networking and sensor networks
  - Software engineering and compilers
  - Intrusion detection
- n Model analysis becomes a necessary part of OS research
- n What is next
  - More OS survivability
  - Pervasive computing
  - Context/Location awareness