Matthew Caesar

483 Soda Hall
Computer Science Division
University of California at Berkeley
Berkeley, CA 94720-1776

mccaesar@cs.berkeley.edu
http://www.cs.berkeley.edu/~mccaesar
Office: 510-289-5078
Fax: 510-642-5775

RESEARCH INTERESTS

Simplifying the management of distributed systems and networks through principles of self-organization and self-diagnosis, with an emphasis on sensornet/wireless networks, overlay networks, and the Internet.

EDUCATION

University of California at Berkeley, *Ph.D. in Computer Science* — Expected May 2007
Dissertation title: *Routing on Flat Identifiers*
Advisors: *Prof. Randy H. Katz, Prof. Ion Stoica*
Minors in Statistics/Machine Learning, CS Theory

University of California at Berkeley, *M.S. in Computer Science* — December 2004
Thesis title: *Root Cause Analysis of BGP Dynamics*
Advisor: *Prof. Randy H. Katz*

University of California at Davis, *B.S. in Computer Science* — May 2000
Senior project: *Resource Management for IP Telephony Networks*
Advisor: *Prof. Dipak Ghosal*

AWARDS AND HONORS

- National Science Foundation (NSF) Graduate Research Fellowship, 2003-2006.
- Department of Defense National Defense Science and Engineering Graduate (NDSEG) Fellowship, 2001-2003.
- US Department of Education Graduate Assistance in Areas of National Need (GAANN) Program Fellowship, 2000.
- UC Davis Computer Science Department Citation Award, 2000.
- Highest GPA among 150+ computer science majors in undergraduate program at UC Davis.
- Rotary Club scholarship (1997), Doyle scholarship (1997).

PROFESSIONAL EXPERIENCE

2001-present **Graduate Student Researcher, RADS project**
Worked under the guidance of Prof. Randy Katz, Prof. Scott Shenker, and Prof. Ion Stoica. I designed, simulated, and implemented an architecture for routing on "flat" identifiers that are free of any location-dependent semantics. I also designed and built an Internet health monitoring system, which inferred root causes of routing anomalies, and leveraged the system to perform a measurement study.

May 2004-Nov 2004 **Research Intern, AT&T Labs, Network Measurement and Engineering group**
Worked with Prof. Jennifer Rexford on the Routing Control Platform (RCP), a system that provides network-wide control for a single ISP network. By directly controlling the routing decisions for a network in real-time, RCP improves the scalability and correctness of route distribution. RCP also simplifies configuration and fault diagnosis by providing a centralized, consistent view of network state. I implemented and co-designed the system, and my work led to a large-scale deployment used in daily network operation in a large tier-1 ISP.

Jun 2003-Sep 2003 **Research Intern/Consultant, Microsoft Research, Systems and Networking**
Worked with Dr. Miguel Castro and Dr. Antony Rowstron on Virtual Ring Routing (VRR), a wireless routing protocol. VRR is a self-organizing protocol for wireless networks that eliminates the need for address configuration by operating directly on flat identifiers as opposed to location-dependent addresses. I co-designed VRR, wrote the first implementation of VRR, and designed several extensions for efficient and correct operation in the context of wireless sensornets. I wrote a simulator based on ns-2, and built and deployed a wireless sensornet implementation of VRR based on TinyOS. Microsoft has incorporated my work as part of the Windows Mesh Connectivity Layer, which is planned to be released as a product.

Apr 2000-May 2001 **Member of Technical Staff / Consultant, iScale Inc.**
Scalable content distribution and streaming media technologies on a Linux platform.

Jun 1999-Sep 1999 **Software Engineering Intern, Hewlett-Packard**
Developed laser biasing algorithm for fiber optic test equipment.

Jun 1998-Sep 1998 **Software Engineering Intern, Nokia**
Worked on hardware for an ATM multiplexer used in a DSL deployment.

Mar 1998-Jun 1998 **Research Intern, Center for Neuroscience, University of California at Davis**
Assisted researchers in designing a medical imaging system for the human brain.

Jun 1997-Sep 1997 **Software Engineering Intern, Diamond Lane Communications**
Worked on software management platform for DSL access multiplexer.

PUBLICATIONS

Routing on flat identifiers:

1. Matthew Caesar, Tyson Condie, Jayanthkumar Kannan, Karthik Lakshminarayanan, Ion Stoica, Scott Shenker, "ROFL: Routing on Flat Labels", *ACM SIGCOMM*, September 2006.
2. Matthew Caesar, Miguel Castro, Edmund Nightingale, Greg O' Shea, Antony Rowstron, "Virtual Ring Routing: Network routing inspired by DHTs", *ACM SIGCOMM*, September 2006.
3. Matthew Caesar, Miguel Castro, Antony Rowstron, "Network Routing", Patent, US 20060039371-pending.

Re-architecting interdomain routing:

1. Matthew Caesar, Jennifer Rexford, "BGP Routing Policies in ISP Networks", IEEE Network Magazine, special issue on Interdomain Routing, Nov/Dec 2005.
2. Lakshminarayanan Subramanian, Matthew Caesar, Cheng Tien Ee, Mark Handley, Morley Mao, Scott Shenker, Ion Stoica, "HLP: A Next-generation Interdomain Routing Protocol", *ACM SIGCOMM*, August 2005.
3. Matthew Caesar, Lakshminarayanan Subramanian, Randy H. Katz, "A Case for an Internet Health Monitoring System", *Hot Topics in System Dependability (HotDep)*, June 2005.
4. Matthew Caesar, Donald Caldwell, Nick Feamster, Jennifer Rexford, Aman Shaikh, Kobus van der Merwe, "Design and Implementation of a Routing Control Platform", *Second Symposium on Networked Systems Design and Implementation (NSDI'05)*, April 2005.
5. Lakshminarayanan Subramanian, Matthew Caesar, Cheng Tien Ee, Mark Handley, Morley Mao, Scott Shenker and Ion Stoica, "HLP: A Next-generation Interdomain Routing Protocol", *Hot Topics in Networking (Hotnets-III)*, November 2004.
6. Matthew Caesar, Lakshminarayanan Subramanian and Randy H. Katz, "Root-cause Analysis of Internet Dynamics", *NANOG 30*, Miami Beach, Florida, February 8-10, 2004.

Streaming media and IP telephony:

1. Bhaskaran Raman, Sharad Agarwal, Yan Chen, Matthew Caesar, Weidong Cui, Per Johansson, Kevin Lai, Tal Lavian, Sridhar Machiraju, Z. Morley Mao, George Porter, Timothy Roscoe, Mukund Seshadri, Jimmy Shih, Keith Sklower, Lakshminarayanan Subramanian, Takashi Suzuki, Shelley Zhuang, Anthony D. Joseph, Randy H. Katz, Ion Stoica, "The SAHARA Model for Service Composition Across Multiple Providers", *International Conference on Pervasive Computing (Pervasive 2002)*, August 2002.
2. Matthew Caesar, Dipak Ghosal and Randy H. Katz, "Resource Management for IP Telephony Networks", *International Workshop on QoS (IWQoS)*, Miami Beach, Florida, May 15-17, 2002.
3. Matthew Caesar, Sujatha Balaraman and Dipak Ghosal, "A Comparative Study of Pricing Strategies for IP Telephony", *IEEE Globecom 2000*, Global Internet Symposium, November 2000.
4. Matthew Caesar and Dipak Ghosal, "IP Telephony", *Encyclopedia of Telecommunications*, John Wiley & Sons (Invited Article).

Papers under submission:

1. Matthew Caesar, Lakshminarayanan Subramanian and Randy H. Katz, "Design and Implementation of an Internet Health Inferencing System", UCB Technical Report No.CSD-04-1356.
2. Jayanthkumar Kannan, Matthew Caesar, Ion Stoica, Scott Shenker, "A Provably Correct Decentralized Structured Underlay".
3. Matthew Caesar, Jayanthkumar Kannan, Ion Stoica, "Towards a Secure Network-layer DHT".
4. Karthik Lakshminarayanan, Matthew Caesar, Thomas Anderson, Scott Shenker, Ion Stoica, "Convergence Free Routing".
5. Brighten Godfrey, Matthew Caesar, Ion Stoica, Scott Shenker, "Stable Interdomain Routing".

TEACHING EXPERIENCE

Aug 2003-Dec 2003	**Teaching Assistant:** *EE122: Computer Networks, taught by Prof. Scott Shenker and Prof. Ion Stoica at University of California at Berkeley.*
Mar 2001-Jun 2001	**Teaching Assistant:** *ECS152B: Computer Networks, taught by Prof. Demet Aksoy at University of California at Davis.*
Jan 2001-Mar 2001	**Teaching Assistant:** *ECS152B: Computer Networks, taught by Prof. Dipak Ghosal at University of California at Davis.*
Mar 2000-Jun 2000	**Teaching Assistant:** *ECS152B: Computer Networks, taught by Prof. Dipak Ghosal at University of California at Davis.*
Jan 2000-Mar 2000	**Teaching Assistant:** *ECS122A: Algorithm Design and Analysis, taught by Prof. Charles Martel at University of California at Davis.*
Feb 1995-May 1997	**Lab Assistant:** CS Department, Santa Rosa Junior College.

Reviewer: Sigcomm 2006, Sigcomm 2005, Sigcomm 2004, Sigcomm 2003, ANCS 2005, IEEE/ACM Transactions on Networking 2005, Elsevier Computer Networks Journal 2006, Infocom 2003, Infocom 2004, ICC 2007.

Member: IEEE, ACM, USENIX.

Mentoring/service: Coadvised Atul Vasu, an intern at the International Computer Science Institute (ICSI). Coadvised Ian Haken, an undergraduate at UC Berkeley, on a research project. Mentored junior Ph.D. students on specific projects. Led guest lectures in several systems-related classes. Regularly organized mock preliminary exam for junior Ph.D. students.

REFERENCES

Professor Randy H. Katz
United Microelectronics Corporation Distinguished Professor
Computer Science Division
637 Soda Hall
University of California at Berkeley
Berkeley, CA 94720-1776
Phone: (510) 642-8778
Email: randy@cs.berkeley.edu

Professor Scott Shenker
Professor, University of California at Berkeley
Director, ICSI Center for Internet Research
1947 Center Street, Suite 600
Berkeley, CA 94704-1198
Phone: (510) 666-2880
Email: shenker@icsi.berkeley.edu

Professor Ion Stoica
Associate Professor, University of California at Berkeley
Computer Science Division
645 Soda Hall
University of California at Berkeley
Berkeley, CA 94720-1776
Phone: (510) 643-4007
Email: istoica@cs.berkeley.edu

Professor Jennifer Rexford
Professor, Princeton University
Computer Science Department
35 Olden Street
Princeton, NJ 08544
Phone: (609) 258-1771
Email: jrex@cs.princeton.edu

Professor Lakshminarayanan Subramanian
Assistant Professor, New York University
Warren Weaver Hall, Room 405
251 Mercer St.
New York, NY 10012
Phone: (212) 998-3011
Email: lakshmi@cs.nyu.edu

RESEARCH SUMMARY

Automating address assignment: The use of location-based addressing in routing protocols is a key bottleneck in building a self-organizing system. It complicates allocation, makes it harder to write network-level access controls, and requires operation and maintenance of a secondary resolution system like DNS. As part of my thesis, I designed and implemented a practical algorithm for routing on flat identifiers. I implemented and evaluated my design for three environments: a wireless sensornet, a small-scale wired enterprise network, and the large-scale Internet. I developed a distributed simulation environment which demonstrated my protocol could scale to networks of hundreds of millions of hosts. I also developed a smaller-scale implementation that was deployed on PlanetLab. In the context of sensornets, I conducted simulation experiments using ns-2, and developed a distributed implementation in which I deployed and evaluated performance in two sensornet testbeds. I also spent several months at Microsoft Research Cambridge where I collaborated with several researchers to deploy my design on their own internal network. During this collaboration, my work was incorporated as the core functionality of a wireless routing protocol soon to be released as part of the mesh connectivity layer in Windows. These deployments were invaluable in driving the design process. For example, in an early deployment on a wireless testbed, I found that link asymmetries arising from unexpected radio interactions could cause the structure of the protocol to converge at a slow rate. To address this, I designed and implemented a mechanism to heal the network in the presence of these asymmetries. This observation led me to formulate an analytical model of the design to further analyze these properties, and I have used the model to prove correctness in the presence of fail-stop failures. My work on automating address assignment led to a patent and two publications in Sigcomm 2006.

Automating network configuration: Routing protocols used today in the Internet are very complex, due to their numerous policy knobs and features. The protocols involved in distributing Internet routes can overlap and conflict in various unpredictable ways, triggering protocol oscillations and forwarding loops. As part of my Ph.D., I built the Routing Control Platform (RCP), a logically centralized platform that performs path selection on behalf of routers in an ISP. The system applies correctness checks before distributing routes, and provides a centralized point at which operators may efficiently configure the network and diagnose problems. I evaluated performance through an implementation followed by an evaluation within a real tier-1 ISP. After several iterations over the design/implementation/optimization cycle, the performance of my implementation reached a point where a single desktop machine running my code was able to keep up with the control traffic of the entire ISP, which was composed of over 500 routers. During this evaluation, I observed that certain events triggering high frequency changes caused substantial loads on routers. To further investigate these events, I designed and implemented an Internet health monitoring system, which localized root causes of routing changes. Given a set of observed updates, the system determines the set of possible ISPs and the set of possible causes that could have generated the set. By narrowing down the cause for an event, such a system can assist network operators in debugging. My system monitored network paths in real time over a period of 24 months. During this work I made the observation that there was a class of failures that was not possible to infer solely from observing path changes. This observation led me to pursue the design of a more efficient and diagnosable inter-domain architecture called HLP. HLP is a hybrid link-state and path-vector replacement for the Internet architecture that provides improved convergence properties and simplifies debugging by exposing link-state information within hierarchies. My work on automating network configuration led to publications in NSDI 2005, Sigcomm 2005, Hotnets 2005, and HotDep 2005.

RESEARCH STATEMENT – Matthew Caesar

The Internet, composed of hundreds of millions of potentially-misbehaving hosts and thousands of competing ISPs, stands as one of the most complex and large-scale systems ever built by humankind. Concerns about the Internet's ability to meet ever-increasing demands on performance and functionality in the presence of this complexity has led to a call to redesign the Internet's architecture, for example in the context of the NewArch project, and NSF's GENI and FIND programs. A key challenge faced in designing a new Internet lies in *management* and *configuration*. These issues were overlooked when designing early data networks and the Internet has been paying a massive price ever since. ISPs hire armies of engineers to manually configure routers and debug problems, and in daily life we are surrounded by an ever-increasing array of complex embedded devices that require substantial configuration to interoperate. Forcing humans to configure and manage networks increases reaction time to faults, introduces the potential for misconfiguration, and substantially increases operating costs.

What is lacking today is a principled look at how to make systems manage themselves. We need a fresh approach to designing networks and protocols with self-management in mind. Toward this goal I propose to develop a class of *self-* protocols* that bootstrap, configure, and troubleshoot problems with only minimal manual intervention. In particular, these protocols aim to *self-configure* in the presence of arbitrary topologies and failure modes, *self-diagnose* routing problems, and *self-tune* operation based on diagnoses.

Prior work

As a first step in this direction, my thesis focused on the particular problem of building a self-managing network protocol. A self-managing network must perform two functions: assign identifiers to nodes and build routes between nodes. Today, network operators are heavily involved in these actions. To assign identifiers, operators must distribute IP address ranges to routers and end hosts, bind DNS names to these addresses, and assign access controls. To build routes, operators must configure policies in underlying routers, check for oscillations or other anomalous behavior arising from violations of policy correctness, and troubleshoot reachability problems. Hence, a self-managing routing protocol must automate both addressing and routing. My work deals with addressing through a new approach to routing that eliminates addressing entirely, and allows routers to forward packets directly on end-host names. To deal with routing, I developed a logically-centralized network-wide control plane that automates router configuration, performs route assignment on behalf of routers, and localizes faults and misconfigurations.

Future work

New network architectures:

New Internet architecture: The Internet is suffering a relentless inflation of routing update loads that shows no signs of slowing. Worse still, recent measurement studies show that this inflation is exceeding the capabilities of Moore's law, requiring nonlinear cost increases in router hardware to keep up. Hence today's architectural requirement of maintaining a complete table of all destination prefixes is quickly becoming impossible, especially in the presence of increased demands for deaggregation and the larger address spaces of IPv6. I have recently begun developing a more scalable network architecture based on my thesis work that does not require a complete table. Instead of routing on flat *host* identifiers, we treat each *subnet prefix* as a numeric identifier. We then leverage a scheme based on my thesis work to self-organize and route between subnets by making progress in the prefix space towards the destination subnet. This approach retains several benefits of flat identifiers, yet has several deployability advantages.

Using randomization to build stable systems: We are developing techniques for improving stability of distributed systems by intelligently selecting which components to use. Our techniques are based on an observation called the *waiting time paradox*, which states that if interarrival times of failures have any variance, the expected waiting time to failure when selecting components randomly is biased towards longer times. We have found that incorporating randomization into component selection can drastically reduce sensitivity to faults. We are developing general techniques that leverage this insight to improve stability in several distributed systems, including path selection in wireless networks, and virtual machine migration in service clusters.

Convergence-free routing: Today's Internet suffers from long outages arising from a slow convergence process that occurs after certain events. Slowing convergence was in fact done by design to improve stability: by rate-limiting updates, fewer route changes take place. Conventional wisdom is that this tradeoff is fundamental. I have recently begun exploring an alternate approach where instead of immediately propagating updates after a failure, we allow routing state to remain inconsistent, but provide a mechanism for data packets to autonomously discover a working path. Preliminary results show this approach reduces churn while maintaining extremely low loss rates.

Debugging/troubleshooting:

Wireless health monitoring: Wireless networks experience a vast array of failure modes which cause instability, starvation, and inefficiency. These problems are notoriously difficult to debug. To address these problems I aim to build a wireless health monitoring system. This tool leverages recent developments in phased array antenna technology to form a complete view of network operation from a single vantage point by passively observing wireless signals. We are currently developing a set of inference techniques that leverage observations from the system to localize problems. Such a tool can be used to diagnose faults, collect insights on how to reconfigure the network to eliminate hotspots, and detect malicious behavior and unauthorized connections.

Learning-based system management: Faults in distributed systems are currently a "fact of life": software has bugs, operators make mistakes, and systems get attacked. However, troubleshooting faults and tuning system performance is currently a black art, relying heavily on heuristics. To formalize the management of systems, I aim to apply machine learning to model, predict, and diagnose system misbehavior. I am interested in applying statistical inference techniques to contain instability and localize faults in networks and component-based applications. My goal is to develop techniques to perform inference quickly, to speed diagnosis and reaction time.

Debugging across competitive domains: A key challenge in troubleshooting today's large-scale distributed systems lies in coordinating the effort across multiple autonomous participants who may distrust each other or have privacy concerns. I am interested in leveraging privacy-preserving query protocols to troubleshoot faults across networks owned by different providers. In this fashion, participants can localize and route around faults without becoming aware of the specific root cause.

TEACHING STATEMENT – Matthew Caesar

I am excited about commencing my career as a professor. Through my experiences in teaching, I have acquired a great deal of energy from communicating with students and managing a classroom. Such interactions reignite my passion and curiosity to explore all aspects of the field of networking. Observing the field through the eyes of a student can lead to questioning fundamental assumptions, which in turn can lead to discovering important problems on which to devote attention. Although teaching and mentoring takes effort and focus to do well, it is also a highly enjoyable process that can benefit the teacher as much as the student.

Teaching philosophies

Learning through doing: It is very important for students to gain experience in working with real systems. Many systems appear simple at first glance, but behave counterintuitively at scale or under certain failure modes. Understanding these behaviors is crucial in building a practical system and refining system properties. For example in my work on self-organizing networks, although the system worked correctly during initial deployment, performance was poor. After observing the protocol in execution, I acquired insights into the design that were difficult or impossible to observe from the formalism of the protocol's state design. These led me to an algorithm with much improved convergence properties. I owe much of my success as a systems builder to the numerous class projects I have been exposed to at Berkeley. Hence as a professor I would strive to formulate challenging yet relevant projects for my students.

Synergy with research: A professor's job is not simply to teach information, but also to teach students how to think critically. I believe forming a strong synergy between research, industrial collaboration, and teaching can be beneficial towards this goal. Integrating students into research helps them gain experience in solving real-world problems, and maintaining strong ties with industry gives feedback that helps maintain a teaching agenda reflecting the state-of-the-art. For example, my time at AT&T gave me insights into the sorts of problems important to the operation of a large tier-1 ISP, which led to the start of collaboration on several research projects. At Microsoft I obtained a different yet complementary view of the issues important to a large software company. Their adoption of one of my algorithms for a product under development led to a considerable amount of feedback and suggestions for improving the deployability of my protocol. As a professor, I would aim to maintain a strong feedback loop across research, teaching, and industrial collaboration by integrating students into my research and industrial collaborations, working jointly with students on projects, supervising student internships, and offering graduate seminar classes.

Inspiration through effective teaching: Building a strong teaching and mentoring agenda can lead to significant payoffs in improving students' understanding and motivation to learn. As a professor, I would aim to maximize my teaching effectiveness in three key ways. First, I consider solid knowledge and constant refinement of course material crucial. During my Ph.D., I served as a teaching assistant five times (four times for the same course). I formulated different projects/homeworks each semester, explored varying topics in my lectures, and otherwise refined my teaching material. I found that the extra time I spent on each class had a significant payoff in improving students' understanding. Second, I consider inspiring and providing motivation for students to learn to be one of the most important aspects of teaching. I feel the greatest gift from my advisors came not from teaching me technical concepts, but rather by instilling within me a strong enthusiasm for solving problems. Hence I typically incorporate into my teaching real-world practical applications of technical concepts, and how such applications vastly benefit society. Finally, I have found it important to adapt to student needs when teaching. Not all students learn the same way, so it is important to isolate a student's strengths when determining research projects, while at the same time giving them an opportunity to broaden their strengths.

Teaching plans

There are a wide variety of classes I would enjoy teaching. Given my thesis, I feel well-prepared to teach classes in distributed systems, operating systems, and networking at the undergraduate and graduate levels. I would also enjoy teaching classes in introductory programming and in related systems areas such as databases and algorithm design. If I were given the opportunity to teach a special-topics course on any topic I chose, it would be one of the following:

Debugging and troubleshooting: This class would explore online analysis and inference of protocols and applications, including what information can be inferred by observing protocols in execution or by observing interdependencies of application components, and how to design protocols and systems to make troubleshooting simple. I would also discuss tradeoffs in inferring root cause in the absence of complete visibility, for example when diagnosing faults in the presence of privacy and security considerations.

Sensornets and embedded systems: One of my primary goals in teaching is to involve students in building and experimenting with the complexities of real systems. Sensornets as a programming environment offer a unique set of challenges, including drastic constraints on memory, power, and processing capabilities. These challenges are becoming increasingly relevant with recent industry focus on building smaller and more functional embedded systems with advanced network capabilities.

Internet architecture: In order to understand the design of a large-scale system such as the Internet, it is necessary to understand its constituent protocols, the problems they solve, and how they interact with policies, network structure, and essential feature requirements that give rise to their design. In this class I would touch on key issues in naming, addressing, routing and management, and overview the wide variety of proposed future architectures and the possible role that the GENI experimental facility might play in exploring those designs.

Summary

In conclusion, I look forward to a career involving teaching due to the wide variety of unique challenges it offers. From my own experiences, I feel that teaching is greatly enhanced via hands-on experience with building systems, maintaining a strong synergy with research, and by demonstrating a strong enthusiasm for the material. I believe my thesis experience best qualifies me to teach systems-related classes, including distributed systems, operating systems, and networking. I would also enjoy teaching related systems-area topics, including security and database systems. I would also enjoy teaching seminar classes on special topics like debugging and troubleshooting, Internet architecture, and sensor/wireless networks.