

Raja R. Sambasivan

4733 Centre Ave., Apt. 5C
Pittsburgh, PA 15213
412-983-1701
rajas AT andrew.cmu.edu
<http://www.ece.cmu.edu/~rajas>

Carnegie Mellon University
2221F CIC
4720 Forbes Ave.
Pittsburgh, PA 15213

OBJECTIVE To obtain a full-time position in the area of computer systems

EDUCATION Ph.D., Electrical & Computer Engineering, May 2012 (expected)
Carnegie Mellon University, Pittsburgh, PA
Advisor: Professor Greg Ganger

M.S., Electrical & Computer Engineering, May 2004
Carnegie Mellon University, Pittsburgh, PA

B.S., Electrical & Computer Engineering w/minor in Computer Science, May 2003
Carnegie Mellon University, Pittsburgh, PA

HONOURS AND AWARDS SIGMETRICS 2007 best paper award (*Modeling the relative fitness of storage*)
Featured in Piled Higher & Deeper ([PhDComics](#)), February 14th, 2007 [strip](#)
FAST 2005 best paper award (*Ursa Minor: versatile cluster-based storage*)

RESEARCH SUMMARY *Research interests:* cloud computing, distributed systems, storage systems, problem diagnosis, tracing and analysis, applying machine learning and visualization techniques to systems problems

SEPT. 2006 PRESENT **Creating tools for automating performance problem diagnosis in distributed systems**
Dissertation topic

Due to their size and complexity, problem diagnosis in large distributed systems and cloud infrastructures is very difficult. Sophisticated tools that use machine learning, statistics, and visualization techniques are needed to help developers with complex diagnosis tasks. My research focuses on using end-to-end tracing, which captures the path of individual requests within and among the components of the distributed environment with very little overhead, to create such tools.

For my dissertation research, I have created a technique called *request-flow comparison*, which automatically localizes the source of performance changes. Such changes are common in cloud infrastructures and so are an important area on which to focus when building automation tools. The key insight behind request-flow comparison is that performance changes often manifest as mutations in the path requests take through the system—e.g., the components they access, the functions they execute—or their timing. Identifying these mutations and showing how they differ from previous behaviour helps localize the source of the problem and significantly guides developer effort. So far, I have shown the effectiveness of this technique by using it to diagnose real, previously undiagnosed performance problems in the Ursa Minor distributed storage service and in certain Google services.

JAN. 2005 MAY 2010 **Self-* Storage: Automating storage management**

Human administration of storage systems is a large and growing issue in modern IT infrastructures. To combat this problem, my research group and I explored ways to create self-configuring, self-organizing, self-tuning, self-healing, and self-managing systems by building the Ursa Minor distributed storage service. My dissertation research is an outgrowth of this work.

- JUN. 2005
MAY 2010
- Creating a transparently scalable metadata service for distributed storage**
Modern object-based storage services, such as the Google File System, are scalable with regards to file data, but because they often only support a single metadata server, not number of files. In this work, we explored the effectiveness of simple techniques for building a scalable metadata service for Ursa Minor. The key challenge we addressed was that of how to minimize the complexity of handling operations, such as RENAMES, that might require coordination between multiple metadata servers. Our approach was twofold. We first minimized such occurrences by automatically placing metadata for files close to one another in the file system on the same metadata server. In cases where the metadata involved resided on multiple servers, we used the migration functionality already present in most systems to move one item to the other item's server. This is a heavy-handed approach, but we found that it worked with little performance impact for many common workloads.
- JUL. 2007
SEPT. 2007
- Improving the accuracy of query-progress indicators for data warehouses**
Due to their complexity, queries in business intelligence workloads often require a large amount of time to complete. As such, it is desirable it to accurately estimate query progress; such information would allow operators to decide whether to kill a currently long-running query, losing all work already done on its behalf, in favour of other more important queries. While an intern at HP Labs, I explored the use of statistical methods for creating a progress indicator for Neoview, HP's data warehousing solution. Our approach involved two steps. First, we utilized linear regression and input transformations to predict the remaining runtime of the individual operators of a query plan. Second, we combined these predictions with knowledge about which operators could execute in parallel to estimate the total remaining runtime.
- JUL. 2005
DEC. 2006
- //TRACE: Enabling accurate trace replay for parallel applications**
Since company security policies often prohibit sharing of applications, it is useful for storage vendors to be able to use trace replay to evaluate their storage systems under a potential client's expected workload. Unfortunately, the many inter-node dependencies exhibited by most parallel applications make them hard to replay accurately. To address this problem, I helped create //TRACE, a program for extracting and replaying traces of parallel applications to recreate their I/O behavior. Its tracing engine automatically discovers inter-node data dependencies and inter-I/O compute times for each node in a parallel application. //TRACE embeds this information in per-node annotated I/O traces, allowing its parallel replayer to closely mimic the behaviour of a traced application.
- JUN. 2003
SEPT. 2005
- Evaluating replication policies for layered clustering of NFS servers**
Layered clustering (or NAS aggregation) offers cluster-like load balancing for unmodified NFS or CIFS servers. Read requests sent to a busy server can be offloaded to other servers holding replicas of the accessed files. In this research, we explored a key design question for this approach: which files should be replicated? By conducting a trace-based study, we found that the popular policy of replicating read-only files offers little benefit. A policy that replicates read-only portions of read-mostly files, however, implicitly coordinates with client cache invalidations and thereby allows almost all read operations to be offloaded. In a read-heavy trace, we found that 75% of all operations and 52% of all data transfers can be offloaded from a busy server.

PROFESSIONAL
EXPERIENCE

Graduate Student

June 2006 – Present

Carnegie Mellon University, Parallel Data Lab

- Currently researching ways to use machine learning, statistics, and visualization techniques to automatically localize problems in cloud infrastructures
- Collaborated with fellow graduate students on the following research topics:
 - Building scalable metadata services for distributed storage systems
 - Building an accurate MPI-based trace replayer for HPC applications
 - Applying machine learning to predict workload performance
- Responsible for developing and maintaining Ursa Minor's NFS server and tracing component

Software Engineering Intern

May 2010 – December 2010

Google

- *Mentors:* Michael De Rosa and Brian McBarron
- Implemented my dissertation research (request-flow comparison) on top of Dapper, Google's end-to-end tracing system, and showed its utility in helping diagnose performance changes
- Helped create a new visualization for Dapper traces that shows service inter-dependencies annotated with performance metrics

Research Intern

July 2007 – March 2008

HP Labs

- *Mentor:* Kivanc Ozonat
- Explored how to create a query progress indicator for Neoview, HP's enterprise data warehouse, via the use of simple statistical techniques
- Created tools for visualizing the complex query execution plans created by Neoview
- Consulted with the statistical learning inference and control (SLIC) team to identify how machine learning could be used to help diagnose performance problems in distributed systems

Systems Programmer

June 2004 – May 2006

Carnegie Mellon University, Parallel Data Lab

- Core member of the Ursa Minor development team
 - Responsible for the development and maintenance of the NFS server and involved with efforts to develop and debug other parts of the system
 - Led an effort to ensure SPEC SFS compatibility with Ursa Minor
 - Developed tools for visualizing Ursa Minor's performance on key benchmarks over time
 - Co-led an effort to improve small-file performance by using intelligent prefetching to eliminate metadata server accesses
- Helped implement a tee that evaluated correctness of a new NFSv3 or NFSv4 server by relaying requests to a known working server and comparing the responses
- Created a NFS trace-based simulator for evaluating NAS replication policies

Student Researcher

January 2003 – May 2003

Carnegie Mellon University, Advanced Multimedia Processing Lab

- Developed and tested a face detection and tracking algorithm for 3D point-cloud cameras

Technical Support Specialist

September 2000 – May 2001

Carnegie Mellon University, Computing Services Help Desk

- Provided technical support for members of the campus community

TEACHING
EXPERIENCE

Teaching Assistant

Fall 2005 & Spring 2010

ECE 18-746, Storage Systems

- Created a project requiring students to build a RAID-5 controller for iSCSI
- Held regular office hours to help students with course material
- Created and graded both tests and projects

RELEVANT CLASSES

18-741: Advanced computer architecture

18-746: Advanced storage systems

15-712: Advanced operating systems & distributed systems

10-701: Graduate machine learning

36-625: Graduate probability and mathematical statistics I

36-626: Graduate probability and mathematical statistics II

15-857: Performance modeling (queuing theory)

SPECIAL SKILLS

Operating systems: Linux, Solaris, Win32, Mac OS X

Programming languages: C, C++, Java, HTML, and Perl

Protocols: NFS (expert), MPI (basic), RPC (advanced), and XDR (advanced)

Data-Intensive Computing paradigms, such as Map Reduce

Extensive knowledge of Matlab, R, and their common toolboxes

AFFILIATIONS

Parallel Data Laboratory (PDL), Carnegie Mellon University

Intel Science and Technology Center for Cloud Computing (ISTC-CC)

Association for Computing Machinery (ACM)

Institute for Electrical and Electronic Engineers (IEEE)

Raja R. Sambasivan

REFEREED PUBLICATIONS

Diagnosing performance changes by comparing request flows. Raja R. Sambasivan, Alice X. Zheng, Michael De Rosa, Elie Krevat, Spencer Whitman, Michael Stroucken, William Wang, Lianghong Xu, Gregory R. Ganger. In proceedings of the 8th USENIX Symposium on Network Systems Design and Implementation (NSDI'11). March 30th to April 1st, 2011. Boston, MA, USA.

A transparently-scalable metadata service for the Ursa Minor storage system. Shafeeq Sinnamohideen, Raja R. Sambasivan, Likun Liu, James Hendricks, Gregory R. Ganger. In proceedings of the 2010 USENIX Annual Technical Conference (USENIX ATC'10). June 23rd to 25th, 2010. Boston, MA, USA.

Categorizing and differencing system behaviours. Raja R. Sambasivan, Alice X. Zheng, Eno Thereska, Gregory R. Ganger. Appears in the proceedings of the 2nd International Workshop on Hot Topics in Autonomic Computing (HotAC II). June 15th, 2007. Jacksonville, Florida, USA.

Modeling the relative fitness of storage. Michael Mesnier, Matthew Wachs, Raja R. Sambasivan, Alice X. Zheng, Gregory R. Ganger. In proceedings of the International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'07). June 12th to 16th, 2007. San Diego, CA, USA.

//TRACE: parallel trace replay with approximate causal events. Michael Mesnier, Matthew Wachs, Raja R. Sambasivan, Julio Lopez, James Hendricks, Gregory R. Ganger. In proceedings of the 5th conference on File and Storage Technologies (FAST'07). February 13th to 16th, 2007. San Jose, CA, USA.

Ursa Minor: Versatile cluster-based storage. Michael Abd-El-Malek, William V. Courtright II, Chuck Cranor, Gregory R. Ganger, James Hendricks, Andrew J. Klosterman, Michael Mesnier, Manish Prasad, Brandon Salmon, Raja R. Sambasivan, Shafeeq Sinnamohideen, John D. Strunk, Eno Thereska, Matthew Wachs, Jay J. Wylie. In the proceedings of the 4th USENIX conference on File and Storage Technologies (FAST'05). December 13th to 16th, 2005. San Francisco, CA, USA.

Replication policies for layered clustering of NFS servers. Raja R. Sambasivan, Andrew J. Klosterman, Gregory R. Ganger. Appears in the proceedings of the 13th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'05). September 27th to 29th, 2005. Atlanta, Georgia, USA.

JOURNAL PUBLICATIONS

Relative fitness modeling. Michael Mesnier, Matthew Wachs, Raja R. Sambasivan, Alice Zheng, Raja R. Sambasivan, Gregory R. Ganger Research Highlights, Communications of the ACM. April 2009.

Early experiences on the journey towards self-* storage. Michael Abd-El-Malek, William V. Courtright II, Chuck Cranor, Gregory R. Ganger, James Hendricks, Andrew J. Klosterman, Michael Mesnier, Manish Prasad, Brandon Salmon, Raja R. Sambasivan, Shafeeq Sinnamohideen, John D. Strunk, Eno Thereska, Matthew Wachs, Jay J. Wylie. In the Bulletin of the IEEE Computer Society Technical Committee on Data Engineering 29(3). Special issue on self-managing database systems. September 2006.

TECHNICAL REPORTS

Automation without predictability is a recipe for failure. Raja R. Sambasivan, Gregory R. Ganger. Carnegie Mellon University Parallel Data Laboratory Technical Report CMU-PDL-11-101. January 2011.

Diagnosing performance changes by comparing system behaviours. Raja R. Sambasivan, Alice X. Zheng, Elie Krevat, Spencer Whitman, Michael Stroucken, William Wang, Lianghong Xu, Gregory R. Ganger. Carnegie Mellon University Parallel Data Laboratory Technical Report CMU-PDL-10-107. July 2010.

A transparently-scalable metadata service for the Ursa Minor storage system. Shafeeq Sinnamohideen, Raja R. Sambasivan, James Hendricks, Likun Liu, Gregory R. Ganger. Carnegie Mellon University Parallel Data Laboratory Technical Report CMU-PDL-10-102. March 2010.

Diagnosing performance problems by visualizing and comparing system behaviours. Raja R. Sambasivan, Alice X. Zheng, Elie Krevat, Spencer Whitman, Gregory R. Ganger. Carnegie Mellon University Parallel Data Lab Technical Report CMU-PDL-10-103. February 2010.

Eliminating cross-server operations in scalable file systems. James Hendricks, Shafeeq Sinnamohideen, Raja R. Sambasivan, Gregory R. Ganger. Carnegie Mellon University Parallel Data Lab Technical Report CMU-PDL-06-105. May 2006.

Improving small file performance in object-based storage. James Hendricks, Raja R. Sambasivan, Shafeeq Sinnamohideen, Gregory R. Ganger. Carnegie Mellon University Parallel Data Lab Technical Report CMU-PDL-06-104. May 2006.

Selected project reports, Spring 2005 Advanced OS & Distributed Systems (15-712). Garth A. Gibson and Hyang-Ah Kim, Editors. Jangwoo Kim, Eriko Nurvitadhi, Eric Chung; Alex Nizhner, Andrew Biggadike, Jad Chamcham; Srinath Sridhar, Jeffrey Stylos, Noam Zeilberger; Gregg Economou, Raja R. Sambasivan, Terrence Wong; Elaine Shi, Yong Lu, Matt Reid; Amber Palekar, Rahul Iyer. Carnegie Mellon Computer Science Technical Report CMU-CS-05-138. May 2005.

Ursa Minor: Versatile cluster-based storage. Michael Abd-El-Malek, William V. Courtright II, Chuck Cranor, Gregory R. Ganger, James Hendricks, Andrew J. Klosterman, Michael Mesnier, Manish Prasad, Brandon Salmon, Raja R. Sambasivan, Shafeeq Sinnamohideen, John D. Strunk, Eno Thereska, Matthew Wachs, Jay J. Wylie. Carnegie Mellon University Parallel Data Laboratory Technical Report CMU-PDL-05-104. April 2005.

PATENTS

Managing execution of database queries. Stefan Kompres, Harumi Anne Kuno, Umeshwar Dayal, Janet Wiener, Raja Sambasivan. *Patent pending.* Filed September 2008.

CONFERENCE
TALKS

Generalizing request-flow comparison to more systems. WiP talk at 23rd ACM Symposium on Operating Systems Principles (SOSP'11). October 2011.

Diagnosing performance changes by comparing request flows. Presented at the 8th USENIX Symposium on Networked Systems Design and Implementation (NSDI'11). March 2011.

Spectroscope: a tool for categorizing and differencing system behaviours. Presented at the 2nd International Workshop on Hot Topics in Autonomic Computing (HotACII). June 2007.

Replication policies for layered clustering of NFS servers. Presented at the 13th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'05). September 2005.

INVITED TALKS

Diagnosing performance changes by comparing request flows. Presented at Google NYC and NetApp RTP. June and September 2011.

Raja R. Sambasivan

REFERENCES

Academic

Gregory R. Ganger

Professor of ECE & CS
Director, Parallel Data Lab
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
ganger AT ece.cmu.edu

Alice Zheng

Researcher
Microsoft Research
One Microsoft Way
Redmond, WA 98052
alicez AT microsoft.com

Rodrigo Fonseca

Assistant Professor
Brown University
Box 1910
115 Waterman St
Providence, RI 02912
rfonseca AT cs.brown.edu

Christos Faloutsos

Professor of CS & ECE
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
mwalgora AT cs.cmu.edu

Internships

Michael De Rosa

Software Engineer
Google
6425 Penn Ave. Suite 700
Pittsburgh, PA 15206
mderosa AT google.com

Kivanc Ozonat

Senior Research Scientist
HP Labs
1501 Page Mill Rd
Palo Alto, CA 94304
kivanc.ozonat AT hp.com