Wyatt Lloyd

http://www.cs.princeton.edu/~wlloyd

205 Hudson St., Apt 601 — wlloyd@cs.princeton.edu
Hoboken, N.J. 07030 — (814) 880-1392

Education

Princeton University .... Princeton, NJ
Ph.D. in Computer Science .... Expected June 2013
M.A. in Computer Science 2009
Advisor: Michael J. Freedman

Pennsylvania State University, University Park .... State College, PA
B.S. in Computer Science with distinction 2007
Schreyer Honors College
Advisor: Thomas F. La Porta

Research Interests

The distributed systems and networking problems that underlie the architecture of large-scale websites, cloud computing, and big data.

Dissertation

2010– **Stronger Consistency and Semantics for Geo-Replicated Storage.** Geo-replicated storage systems provide the backend for massive-scale websites such as Twitter and Facebook, storing data that includes your profile, friends lists, and status updates. These storage systems seek to provide an "always-on" experience where operations always complete quickly, because of a widely demonstrated link between page load times, user engagement, and revenue. We term systems that can handle data of this scale and provide the always-on experience *ALPS systems*, because they provide four key properties—availability, low latency, partition tolerance, and scalability.

Our COPS [2] system is the first distributed data store to guarantee the ALPS properties and achieve consistency stronger than eventual. Eventual consistency specifies only that writes in one datacenter eventually show up in the others. Causal consistency, which is what COPS provides, maintains the partial order over operations establish by potential causality. Under causal consistency, all of a user's operations appear in the order they are issued and interactions between users, e.g., conversations in comments, appear in their correct order as well. This improvement in consistency gives users a better experience and makes the data store easier for programmers to reason about. A key technical contribution of the COPS work is its fully distributed and scalable architecture that uses explicit metadata and off-path dependency checks to enforce ordering instead of relying on any single point of coordination.

Our Eiger [1] system further pushes on the semantics an ALPS data store can provide. Eiger provides high-performance, guaranteed low-latency read-only and write-only transactions across the thousands of machines in a cluster. Read-only transactions allow a client to observe a consistent snapshot of an entire cluster. Write-only transactions allow clients to atomically write many values spread across many servers at a single point in time. One important use case for write-only transaction is maintaining symmetrical relationships, e.g., Alice "isAFriendOf" Bob and Bob "isAFriendOf" Alice should both appear or disappear at the same time. Eiger also improves the semantics of ALPS data stores by providing the column-family data model—which is used in BigTable and Cassandra, and can be used to built real applications like Facebook—instead of the key-value data model provided by COPS—which is useful mainly as an opaque cache.

My dissertation research shows that ALPS systems do not need to settle for eventual consistency and weak semantics. Taken together, Eiger and COPS show that causal consistency and stronger semantics are possible for low-latency geo-replicated storage.

Other Research Experience

2009–2011 **Low-Overhead Transparent Recovery for Static Content.** Client connections to web services break when the particular server they are connected to fails or is taken down for maintenance. We designed and built TRODS [3], a system that transparently recovers connections to web services that delivers static content, e.g., photos or videos. TRODS is implemented as a server-side kernel module for immediate deployability, it works with unmodified services and clients. The key insight in TRODS is its use of cross-layer visibility and control: It derives reliable storage for application-level state from the mechanics of the transport layer. In contrast with more general recovery techniques, the overhead of TRODS is minimal. It provides throughput-per-server competitive with unmodified HTTP services, enabling recovery without additional capital expenditures.

2007–2010 **Using History for High-Throughput Fault Tolerance.** Byzantine fault-tolerant (BFT) replication provides protection against arbitrary and malicious faults, but its performance does not scale with cluster size. We designed and built Prophecy [4], a system that interposes itself between clients and any replicated service to scale throughput for read-mostly workloads. Prophecy relaxes consistency to delay-once linearizability so it can perform fast, load-balanced reads when results are historically consistent, and slow, replicated reads otherwise. This dramatically increases the throughput of replicated services, e.g., the throughput of a 4 node Prophecy web service is ~4X the throughput of a 4 node PBFT web service.

2007 **IP Address Passing for VANETs.** In Vehicular Ad-hoc Networks (VANETs), vehicles have short connection times when moving past wireless access points. The time required for acquiring IP addresses via DHCP consumes a significant portion of each connection. We reduce the connection time to under a tenth of a second by passing IP addresses between vehicles. Our implementation improves efficiency, reduces latency, and increases vehicle connectivity without modifying either DHCP or AP software [5].

2006–2007 **Multi-Class Overload Controls for SIP Servers.** When SIP servers that are used for signaling in VoIP network are overloaded, call-setup latency increases significantly and critical calls—e.g., 911 calls—can be denied. My undergraduate thesis on multi-class overload controls [6] reduces call latency by suppressing retransmissions and prioritizes critical calls so they always connect.

Professional Experience

9/07– **Research Assistant.** Princeton University, Princeton, NJ
Major projects include providing stronger consistency for scalable storage systems (COPS), providing stronger semantics for scalable storage systems (Eiger), enabling transparent connection recovery for web services (TRODS), and using history for high-throughput fault tolerance (Prophecy).

5/12–8/12 **Ph.D. Intern.** Facebook, New York, NY
Worked on a distributed-storage-systems team on a project to improve caching for static content. Was the first intern at the new New York office.

6/10–9/10 **Summer Research Fellow.** Intel Labs Pittsburgh / CMU, Pittsburgh, PA
Began leading the COPS project, a collaborative effort between Princeton University, Intel Labs, and Carnegie Mellon University.

6/07–9/07 **Intern-Student Engineer.** The Boeing Company, Anaheim, CA
Worked on an internal research and development project on routing in multi-tier wireless networks as part of the Network Systems group.

5/06–9/06 **Intern-Student Engineer.** The Boeing Company, Anaheim, CA
Worked on an internal research and development project that utilized DHCP for intra-domain mobility management as part of the Network Systems group.

Teaching Experience

11/29/12 **Guest Lecturer.** Distributed Systems, (CMU) 15-440
Lectured on COPS to introduce cutting-edge research to undergraduates.

10/4/12 **Guest Lecturer.** Advanced Computer Networks, COS-561
Lectured on inter-domain routing with BGP and led a discussion of research papers on network isolation and software-defined networking.

2/09–6/09 **Teaching Assistant.** Computer Networks, COS-461
Graded, held office hours, helped design exams, and taught exam-review sessions.

9/08–1/09 **Teaching Assistant.** General Computer Science, COS-126
Graded, held office hours, helped design exams, and taught twice-weekly recitations.

Service

11/9/12 **Panelist.** Princeton Women in Computer Science Graduate School Panel
Shared experiences and advice about graduate school.

10/11–	**Regional Lead.** Organized events for Princeton region and served on the advisory board.	Siebel Scholars Foundation
6/11–8/11 6/09–8/09	**Student Advisor.** Advised novice undergraduate programmers on 6-week-long projects.	Princeton Summer Programming Experience
2/06–5/07	**Student Representative.** Helped shape undergraduate Computer Science curriculum.	Penn State CSE Curriculum Committee

Honors

2012	Wu Prize for Excellence (Princeton)
2012	Facebook Fellowship Finalist
2012	Siebel Scholar
2007	Princeton University Graduate Fellowship
2003-2007	Dean's List (Penn State)
2003-2007	Schreyer Honors College Scholar (Penn State)
2006	College of Engineering General Scholarship (Penn State)
2003	Maryland State Distinguished Scholar
2002	National Merit Scholarship Honorable Mention
2000	Eagle Scout

Refereed Conference Publications

[1] **Wyatt Lloyd**, Michael J. Freedman, Michael Kaminsky, and David G. Andersen. Stronger Semantics for Low-Latency Geo-Replicated Storage. To appear in *Proc. 10th Symposium on Networked Systems Design and Implementation (NSDI 13)*, April 2013. 14 pages.

[2] **Wyatt Lloyd**, Michael J. Freedman, Michael Kaminsky, and David G. Andersen. Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. In *Proc. 23rd ACM Symposium on Operating Systems Principles (SOSP 11)*, October 2011. 16 pages.

[3] **Wyatt Lloyd** and Michael J. Freedman. Coercing Clients into Facilitating Failover for Object Delivery. In *Proc. 41st IEEE/IFIP International Conference on Dependable Systems and Networks, Dependable Computing and Communication Symposium (DCCS) track (DSN 11)*, June 2011. 12 pages.

[4] Siddhartha Sen, **Wyatt Lloyd**, and Michael J. Freedman. Prophecy: Using History for High-Throughput Fault Tolerance. In *Proc. 7th Symposium on Networked Systems Design and Implementation (NSDI 10)*, April 2010. 16 pages.

[5] Todd Arnold, **Wyatt Lloyd**, Jing Zhao, and Guohong Cao. IP Address Passing for VANETs. In *Proc. 6th IEEE International Conference on Pervasive Computing and Communications (PERCOM 08)*, March 2008. 10 pages.

Theses

[6] **Wyatt Lloyd**. Multi Class Overload Controls for SIP Servers. *Honors Thesis*, The Pennsylvania State University, May 2007.

Refereed Conference Presentations

[7] Stronger Semantics for Low-Latency Geo-Replicated Storage. To appear at *10th Symposium on Networked Systems Design and Implementation (NSDI 13)*, April 2013.

[8] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. In *23rd ACM Symposium on Operating Systems Principles (SOSP 11)*, October 2011.

[9] Coercing Clients into Facilitating Failover for Object Delivery. In *41st IEEE/IFIP International Conference on Dependable Systems and Networks, Dependable Computing and Communication Symposium (DCCS) track (DSN 11)*, June 2011.

Other Presentations

[10] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. Facebook Ph.D. Intern and Distributed Systems Reading Group Talk, August 2012.

[11] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. Berkeley, Cloud Seminar, April 2012.

[12] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. Intel Science and Technology Center on Cloud Computing Retreat, Research Talk, December 2011.

[13] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. University of Maryland, SysChat Group Talk, October 2011.

[14] Don't Settle for Eventual: Scalable Causal Consistency for Wide-Area Storage with COPS. Johns Hopkins University, Computer Science Seminar, October 2011.

References

Prof. Michael J. Freedman
Assistant Professor
Computer Science Department
Princeton University
`mfreed@cs.princeton.edu`

Prof. David G. Andersen
Associate Professor
Computer Science Department
Carnegie Mellon University
`dga@cs.cmu.edu`

Dr. Michael Kaminsky
Senior Research Scientist
ISTC for Cloud Computing
Intel Labs
`michael.e.kaminsky@intel.com`

Prof. Mike Dahlin
Professor
Computer Science Department
The University of Texas at Austin
`dahlin@cs.utexas.edu`