

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

PARTEC CLUSTER COMPETENCE CENTER GMBH,
Patent Owner.

IPR2025-00318
U.S. Patent No. 11,537,442
Issued: December 27, 2022
Application No. 16/963,749
Filed: January 23, 2019

Title: Application Runtime Determined Dynamical
Allocation of Heterogeneous Compute Resources

DECLARATION OF DARRELL LONG

MICROSOFT CORP.
EXHIBIT 1003

TABLE OF CONTENTS

	Page(s)
I. INTRODUCTION AND ENGAGEMENT	4
II. BACKGROUND AND QUALIFICATIONS.....	4
III. MATERIALS CONSIDERED AND INFORMATION RELIED UPON REGARDING '442 PATENT	7
IV. UNDERSTANDING OF PATENT LAW	9
V. SUMMARY OF OPINIONS	13
VI. STATE OF THE ART.....	13
VII. LEVEL OF ORDINARY SKILL IN THE ART.....	17
A. Publications Reflecting The Knowledge Of A POSITA	19
VIII. THE '442 PATENT.....	21
A. The '442 Patent's Specification	21
B. The Prosecution History	23
IX. CLAIM CONSTRUCTION	24
X. CLAIMS 1-10 ARE OBVIOUS IN VIEW OF LIPPERT AND BUDENSKE	24
A. Lippert (EX1004)	24
B. Budenske (EX1005)	26
C. Combining Lippert And Budenske	27
1. Claim 1	36
2. Claim 2	45
3. Claim 3	47
4. Claim 4	53
5. Claim 5	53

6.	Claim 6.....	55
7.	Claim 7.....	59
8.	Claim 8.....	60
9.	Claim 9.....	61
10.	Claim 10.....	64
XI.	CLAIMS 2-5 AND 8-10 ARE OBVIOUS IN VIEW OF LIPPERT, BUDENSKE, AND KAMBATLA.....	64
A.	Kambatla (EX1006)	65
B.	Combining Lippert, Budenske, And Kambatla.....	67
1.	Claim 2	73
2.	Claim 3	76
3.	Claim 4.....	79
4.	Claim 5	79
5.	Claim 8.....	81
6.	Claim 9.....	82
7.	Claim 10.....	84

I, Darrell Long, do hereby declare as follows:

I. INTRODUCTION AND ENGAGEMENT

1. I have been retained as an independent expert on behalf of Microsoft Corp. in connection with the above-captioned Petition for *Inter Partes* Review (“IPR”) to provide my analyses and opinions on specific technical issues related to U.S. Patent No. 11,537,442 (hereinafter “the ’442 patent”).

2. I am being compensated at my usual and customary rate for my time with this IPR. My compensation is not affected by the outcome of this IPR.

3. Specifically, I have been asked to provide my opinions regarding whether claims 1-10 (each a “Challenged Claim” and collectively the “Challenged Claims”) of the ’442 patent would have been obvious to a person having ordinary skill in the art (“POSITA”) as of January 2018. It is my opinion that claims 1-10 would have been obvious to a POSITA after reviewing the prior art discussed herein.

II. BACKGROUND AND QUALIFICATIONS

4. I am an expert in the field of distributed computing systems and the fields of computer science and computer engineering, more generally. For over 40 years, I have been working, teaching, or conducting research in the field of computer science and engineering. In formulating my opinions, I have relied upon my training, knowledge, and experience in the relevant art. A copy of my curriculum vitae is appended to this declaration as Appendix A and describes my professional

experience, including my academic and employment history, publications, conference participation, awards and honors, and more. The following paragraphs briefly summarize my relevant qualifications and professional experience.

5. I am a Distinguished Professor of Engineering, *Emeritus*, and a Distinguished Research Professor of Engineering at the University of California, Santa Cruz. I am also currently a Distinguished Visting Scholar at Santa Clara University. I received my B.S. degree in Computer Science from San Diego State University in 1984 and my M.S. and Ph.D. in Computer Science from the University of California, San Diego, respectively, in 1986 and 1988. I joined the faculty of the University of California, Santa Cruz, as an Assistant Professor of Computer Science in 1988. I earned tenure in 1994 and advanced to the rank of Associate Professor and in 1999 to the rank of full Professor, culminating in 2017 as Distinguished Professor of Engineering. I also twice served as Associate Dean for graduate studies and research and the Director of an NSF-funded research center.

6. In addition, I have been Professeur Invité at the Conservatoire National des Arts et Métiers, the Université Paris–Dauphine and Université Paris–Descartes and Sorbonne Université in Paris, France. I am Professor *ad Honorem* de la Universidad Católica del Uruguay and a visiting professor at the University of Technology, Sydney, Australia, and the United States Naval Postgraduate School. I am a European Organization for Nuclear Research (CERN) Associate Member.

7. Throughout my career, I have been actively involved in research. I have received over 50 peer-reviewed grants to fund my computer science and engineering research. I have authored two books and four chapters of books. I have also written 32 journal articles, 155 peer-reviewed conference papers, and 33 peer-reviewed short papers in computer science, including many papers related to distributed computing.

8. I have taught undergraduate and graduate courses on programming and programming languages, operating systems, storage systems, computer systems, cryptography and computer security, data structures, compiler design, and computer networks.

9. In addition to my academic positions, I have served on several special and standing committees of the National Research Council and the Science & Technology Committee for the Livermore and Los Alamos National Laboratories for over a decade. I served on the advisory board in Computer and Information Sciences for Sandia National Laboratory and the Global Security advisory board for Pacific Northwest National Laboratory. I was the Editor-in-Chief emeritus of Letters of the IEEE Computer Society, and am the Editor-in-Chief emeritus of the journal ACM Transactions on Storage, a Fellow of the Institute of Electrical and Electronics Engineers (“IEEE”), and a Fellow of the American Association for the Advancement of Science (“AAAS”).

10. I am a named inventor on 11 U.S. Patents within the field of computer science, with a twelfth patent now being prosecuted.

11. I was a general chair for the conference on Mass Storage Systems and Technologies in 2024. I have been on multiple program committees and steering committees, including committees for the Symposium on File and Storage Systems Technology (a conference that I founded in 2002), the International Conference on Cloud Computing and Services Science, Symposium on Modeling, IEEE, International Conference on Cloud and Big Data Computing, Petascale Data Storage Workshop, and International Conference on Advanced Communications and Computation among many others.

III. MATERIALS CONSIDERED AND INFORMATION RELIED UPON REGARDING '442 PATENT

12. In preparing this declaration and arriving at my opinions, I have reviewed the following materials bearing Exhibit Nos. that I understand are being referenced in an IPR proceeding where my declaration may be referenced:

LIST OF EXHIBITS

No.	Description
1001	U.S. Patent No. 11,537,442 (“the ’442 patent”)
1002	File History of U.S. Patent No. 11,537,442
1003	N/A
1004	U.S. Patent App. Pub. No. U.S. 2013/0282787 (“ Lippert ”)

No.	Description
1005	John R. Budenske et al., <i>A Method for the On-Line Use of Off-Line Derived Remappings of Iterative Automatic Target Recognition Tasks onto a Particular Class of Heterogeneous Parallel Platforms</i> , The Journal of Supercomputing, Vol. 12, No. 4 (Oct. 1998) (“ Budenske ”)
1006	U.S. Patent App. Pub. No. U.S. 2018/0074855 (“ Kambatla ”)
1007	Yu-Kwong Kwok et al., <i>A Semi-Static Approach to Mapping Dynamic Iterative Tasks onto Heterogeneous Computing Systems</i> , Journal of Parallel & Distributed Computing (2005) (“ Kwok ”)
1008	Mitchell D. Theys et al., <i>Mapping Tasks onto Distributed Heterogeneous Computing Systems Using a Genetic Algorithm Approach</i> , in Solutions to Parallel and Distributed Computing Problems (A. Zomaya, F. Ercal & S. Olariu eds., 2001) (“ Theys ”)
1009	U.S. Patent No. 3,496,551 (“ Driscoll ”)
1010	Yuan-Chieh Chow and Walter H. Kohler, <i>Models for Dynamic Load Balancing in a Heterogeneous Multiple Processor System</i> , IEEE Transactions on Computers, Vol. C-28, No. 5 (May 1979) (“ Chow ”)
1011	David M. Nicol and Paul F. Reynolds, Jr., <i>Optimal Dynamic Remapping of Data Parallel Computations</i> , IEEE Transactions on Computers, Vol. 39, No. 2 (Feb. 1990) (“ Nicol ”)
1012	Arun C. Murthy et al., Apache Hadoop YARN (2014) (“ Murthy ”)
1013	Morris Jette and Mark Grondona, <i>SLURM: Simple Linux Utility for Resource Management</i> , ClusterWorld Conference and Expo (2003) (“ Jette ”)
1014	Eicker et al., <i>The DEEP Project: An Alternative Approach to Heterogeneous Cluster-Computing in the Many-Core Era</i> , Concurrency and Computation: Practice and Experience (2016) (“ Eicker ”)

No.	Description
1015	U.S. Patent App. Pub. No. U.S. 2017/0262319 (“ Newburn ”)
1016	Carsten Clauss et al., <i>Dynamic Process Management with Allocation-Internal Co-Scheduling Towards Interactive Supercomputing</i> , Proceedings of the First Workshop on Co-Scheduling of HPC Applications (2016) (“ Clauss ”)
1017	N/A
1018	Samik Raychaudhuri, <i>Introduction to Monte Carlo Simulation</i> , Proceedings of the 2008 Winter Simulation Conference (2008) (“ Raychaudhuri ”)
1019	David W. Bauer, Jr. and Mojdeh Mohtashemi, <i>An Application of Parallel Monte Carlo Modeling for Real-Time Disease Surveillance</i> , Proceedings of the 2008 Winter Simulation Conference (2008) (“ Bauer ”)
1020	List of Challenged Claims
1021	U.S. Patent No. 5,881,227 (“ Brenner ”)
1022	U.S. Patent No. 7,047,299 (“ Curtis ”)
1023	N. Metropolis, <i>The Beginning of the Monte Carlo Method</i> , Los Alamos Science, Special Issue 125-130 (1987) (“ Metropolis ”)

IV. UNDERSTANDING OF PATENT LAW

13. I am not an attorney. For this declaration, I have been informed about certain aspects of the law that are relevant to my opinions. The Petitioner’s attorneys provided me with my understanding of the law.

14. I understand that when considering the scope of the claims of a patent, the patent claim terms should generally be given the ordinary meaning that the terms

would have to a person of ordinary skill in the art in question after reading the patent as of the earliest claimed priority date.

15. I understand that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the term appears but in the context of the entire patent, including the other claims and the application specification as filed. I further understand that the principal considerations regarding the claims' scope and meaning are the claims' plain language (including the surrounding claim language and context), the patent specification, and the patent's prosecution history in the Patent Office. I understand that while a claim is to be read in light of the specification, one must generally avoid importing limitations into the claim from the specification. I am also informed that the prosecution history can often inform the meaning of the claim by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be. I applied these understandings when considering the scope and meaning of the claims of the '442 patent.

16. I further understand that a claim is unpatentable if it would have been obvious. Obviousness of a claim requires that the claim would have been obvious from the perspective of a POSITA at the time the alleged invention was made. I understand that a claim could have been obvious from a single prior art reference or

from a combination of two or more prior art references. I understand that an obviousness analysis requires understanding the scope and content of the prior art, any differences between the alleged invention and the prior art, and the level of ordinary skill in evaluating the pertinent art.

17. I further understand that a claim would have been obvious if it unites old elements with no change to their respective functions, or merely substitutes one element for another known in the field, and that combination yields predictable results. While it may be helpful to identify a reason for this combination, I understand there is no strict requirement to find an express teaching, suggestion, or motivation to combine within the references. When a product is available, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a POSITA can implement a predictable variation, obviousness likely prevents its patentability. For the same reason, if a technique has been used to improve one device and a POSITA would recognize that it would improve similar devices in the same way, using the technique would have been obvious. I understand that a claim would have been obvious if common sense directs one of ordinary skill, without hindsight bias caused by knowledge of the challenged claims, to combine multiple prior art references or add missing features to reproduce the alleged invention recited in the claims.

18. I further understand that certain factors may support or rebut the obviousness of a claim. I understand that such secondary considerations include, among other things, commercial success of the patented invention, skepticism of those having ordinary skill in the art at the time of invention, unexpected results of the invention, any long-felt but unsolved need in the art that was satisfied by the alleged invention, the failure of others to make the alleged invention, praise of the alleged invention by those having ordinary skill or expertise in the art, and copying of the alleged invention by others in the field. I understand that there must be a nexus—a connection—between any such secondary considerations and the alleged invention. I also understand that contemporaneous and independent invention by others is a secondary consideration tending to show obviousness.

19. I am not aware of any allegations by the named inventors of the '442 patent or any assignee of the '442 patent that any secondary considerations would rebut the obviousness of any Challenged Claim of the '442 patent.

20. I understand that in considering obviousness, it is important not to determine obviousness using the benefit of hindsight derived from the descriptions found in the patent being considered.

21. I understand that Petitioner has the burden of proving obviousness by a preponderance of evidence, which means that the claims are more likely than not obvious.

22. The analysis in this declaration is in accordance with the above-stated legal principles.

V. SUMMARY OF OPINIONS

23. In forming the opinions set forth herein, I have considered:

- a. The documents listed above;
- b. My education, knowledge, skills, and experience in the field of computer science and engineering; and
- c. The level of skill of a person having ordinary skill in the art (POSITA) at the time of the alleged invention of the '442 patent, namely January 23, 2018.

24. As I explain in further detail below, it is my professional and expert opinion that claims 1-10 of the '442 patent are obvious over the prior art.

VI. STATE OF THE ART

25. A heterogeneous computing system is a computer architecture that uses multiple types of processors, each better at performing certain particular tasks, and operating systems. *E.g.*, EX1008, 135. POSITAs have long known how to assign tasks between multiple processors for multi-processor systems and heterogeneous computing systems. *E.g.*, EX1009, 1:70-2:4 (as of 1967, “the attention of systems designers is turning ever increasingly towards the concept of multi-processing computing systems wherein a particular large job may be broken up into individual

segments or tasks and performed in parallel in two or more separate processors wherever possible”); EX1010, 354 (describing scheduling tasks to dynamically load balance a “heterogeneous multiple processor system” in 1979).

26. A well-known “design goal of any multi-processor computing system is to achieve high overall efficiency.” EX1009, 2:13-20. To achieve overall efficiency throughout processing of a task, POSITAs have long known of benefits of assigning tasks to processors *dynamically* because “[s]ome computations dynamically change their behavior as they progress.” EX1011, 206.

27. One known type of distributed computing architecture is a “cluster,” which is a collection of interconnected computing devices (“nodes”) utilized as a single, unified computing resource. EX1022, 1:16-54 (describing clusters). Clusters often use “daemons,” a daemon being a background process that performs a specific task¹. EX1021, 7:66-67 (a “daemon” is defined as “a program that runs unattended to perform a standard service”), 8:2-3 (“Daemon processes generally provide services that must be available at all times.”), 2:54-3:18 (describing using daemons to facilitate communication in a “parallel processing environment” comprising “a

¹ The term “daemon” is commonly used in computer science to refer to background processes, inspired by Maxwell’s daemon of physics and thermodynamics. *See* <https://ei.cs.vt.edu/~history/Daemon.html>.

plurality of nodes connected through a network to one another”); EX1022, 3:63-66 (“As is commonly known, a daemon is a process that runs in the background and performs a specified operation at predefined times or in response to certain events.”), 3:66-4:14 (explaining that “[t]ypical daemon processes ... perform administrative tasks” and are implemented in “any environment ... which is capable of implementing background processes and supports communications between the various Nodes of the Cluster(s)”). As of the earliest alleged filing date, POSITAs typically used cluster resource management frameworks such as Hadoop YARN or SLURM to manage the allocation of jobs to cluster resources. EX1012, xvii (explaining that Hadoop is “a compelling part of a modern data architecture for a very wide spectrum of the industry”), xviii (explaining that “YARN makes the effort of building such [distributed] systems significantly simpler by dealing with many issues with which a framework developer would be confronted”), 1-20 (providing a historical overview of Hadoop YARN), 43-58 (providing a “Functional Overview of YARN Components” for managing cluster resources); EX1013, 3 (explaining that “Simple Linux Utility for Resource Management (SLURM) is a resource management system suitable for use on large and small Linux clusters”), 4 (explaining that “[a]s a cluster resource manager, SLURM ... allocates exclusive and/or non-exclusive access to resources (compute nodes) to users” and “provides a framework for starting, executing, and monitoring work (normally a parallel job) on

the set of allocated nodes”). Such frameworks may include a daemon at every cluster node to enable monitoring and control of the node by a central resource management daemon. EX1012, 9 (describing a central daemon and node daemons or “TaskTrackers” for executing tasks at the node), 38-40 (describing a “NodeManager” daemon at each node “responsible for launching the applications’ containers, monitoring their resource usage ..., and reporting the same to the ResourceManager”); EX1013, 5-7 (describing a central controller daemon and daemons running at each compute node “to export control to SLURM”), 20 (describing a daemon at each node “for managing user jobs and monitoring system state,” where its “most common action ... is to report system state on request”).

28. The ’442 patent includes various acknowledgments regarding the state of the art. For example, the ’442 patent acknowledges that heterogeneous computing systems comprising a plurality of computation nodes and a plurality of booster nodes were known. EX1001, 1:23-27. The ’442 patent acknowledges that prior art reference Eicker (EX1014), describes such a heterogeneous computing system capable of “taskifying” an application into a plurality of tasks that are distributed among the nodes to enable scalability. EX1001, 1:39-55. The ’442 patent acknowledges that Newburn (EX1015) describes “runtime processes that can fully or partially automate the distribution of data and mapping of tasks to computing resources.” EX1001, 1:56-58; *see also* EX1001, 5:10-24. The ’442 patent further

acknowledges that resource managers for “dynamic process management” were known and “described in Clauss” (EX1016), a prior art reference which also discloses “daemons as addressed” in the ’442 patent. EX1001, 1:28-35, 3:13-17. As an example of a job with tasks that may be repeated during execution, the ’442 patent suggests a “‘Monte-Carlo’ based simulation,” a well-known method in the art that was also known for efficient processing with cluster computing systems. EX1001, 3:44-49; EX1018, 91 (providing an “introductory tutorial on Monte Carlo simulation, a type of simulation that relies on repeated random sampling and statistical analysis to compute the results”); EX1019, 1029-1031 (describing a cloud computing architecture for “opportunistic” scheduling of “parallel monte carlo modeling methods” among compute nodes of the system), 1032-1036 (discussing scalability of Monte Carlo methods and demonstrating the “highly efficient parallel computation for monte carlo models”); EX1023, 125-130 (discussing the development of the Monte Carlo method for early computers in the 1940s, and the state of parallel processing in 1987 for using the method).

VII. LEVEL OF ORDINARY SKILL IN THE ART

29. I understand that in evaluating obviousness of the ’442 patent claims, the content of a patent or printed publication prior art should be interpreted the way a person of ordinary skill in the art would have interpreted the prior art as of the effective filing date of the challenged patent claims. I assume that the effective filing

date is the date of the European patent application to which the '442 patent alleges priority, namely January 23, 2018. My understanding of a POSITA would not materially differ if the effective filing date were the PCT filing date or the U.S. application date, namely January 23, 2019, or July 21, 2020.

30. I understand that factors that may be considered in determining the level of ordinary skill in the art at the time of the effective filing date of the challenged patent include: (1) the educational level of the inventor; (2) the types of problems encountered in the art; (3) prior art solutions to those problems; (4) rapidity with which innovations are made; (5) sophistication of the technology; and (6) educational level of active workers in the field.

31. A person of ordinary skill in the art ("POSITA") would include someone who had, in January 2018, (i) a bachelor's degree in computer science, electrical engineering, computer engineering, or a closely related field, as well as (ii) approximately three years of industry experience or research with distributed computing technologies. Additional education could substitute for the industry experience as it typically involves at least 2-4 years of additional study focused exclusively on computer science, as well as a thesis or project spanning several terms to implement a substantial project.

32. Such industry experience working with distributed computing technologies would have included experience designing, implementing, and

maintaining distributed computing systems, such as cluster computing systems, installing and adapting cluster resource management frameworks on distributed computing systems, and deploying applications on such systems, as described above in the State Of The Art. A POSITA would have known of the basic design principles of fault tolerance and efficient scheduling/task assignments.

33. I would have met the requirements of a POSITA in the period from January 23, 2018, to July 21, 2020. Some of my past students would also have qualified as POSITAs on these dates.

A. Publications Reflecting The Knowledge Of A POSITA

34. The knowledge of a POSITA is reflected in well-known and widely available books and industry publications, which confirm my own opinions as to the knowledge of a POSITA. These books and industry publications, described below, would have been known and readily available to POSITAs in January 2018.

35. Budenske (EX1005) published in 1998 in the well-regarded Journal of Supercomputing, which has been publishing papers on the technology and applications of supercomputing since 1987. Budenske describes dynamically re-mapping sub-tasks among processors of a heterogeneous computing system between iterations in order to minimize the execution time of each iteration.

36. Kwok (EX1007) published in 2005 by Elsevier in the Journal of Parallel and Distributed Computing expands the Budenske methodology by evaluating

techniques to associate dynamic parameters with mappings. Theys (EX1008) published in 2001 by Wiley, a well-known publisher, as part of the book *Solutions to Parallel and Distributed Computing Problems*, which covers biology-inspired techniques for computing (e.g., neural networks, genetic algorithms). Theys describes the Budenske methodology, with a focus on using genetic algorithms to map sub-tasks to processors optimally.

37. Chow (EX1010) and Nicol (EX1011) were both published in the journal *IEEE Transactions on Computers*, a well-respected journal widely known in the art, in 1979 and 1990 respectively.

38. Murthy (EX1012) was published in 2014 and is a well-known text on Apache Hadoop YARN written by its creators that describes how to install and administer clusters with YARN, as well as how to develop applications and frameworks with YARN.

39. Jette (EX1013) published in 2003 and is an overview of SLURM, the “open source, fault-tolerant, and highly scalable cluster management and job scheduling system for Linux clusters of thousands of nodes.” EX1013, 3.

40. Eicker (EX1014) published on June 10, 2016, by Wiley InterScience in the journal *Concurrency and Computation: Practice and Experience*.

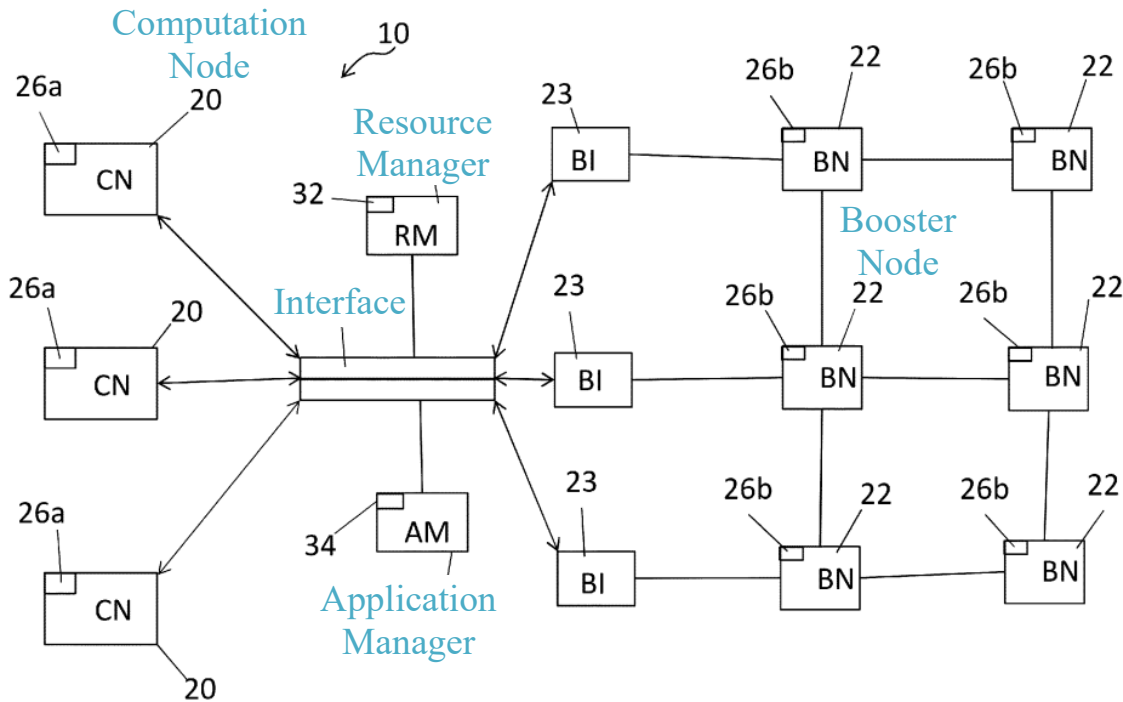
41. Clauss (EX1016) was presented and published on January 19, 2016, in the First Workshop on Co-Scheduling of HPC Applications in Prague, Czech Republic. EX1016, 1.

42. Raychaudhuri (EX1018) and Bauer (EX1019) were presented at the 2008 Winter Simulation Conference in Miami, Florida, and published in the Proceedings of the 2008 Winter Simulation Conference. *E.g.*, EX1018, i, 91; EX1019, i, 1029.

VIII. THE '442 PATENT

A. The '442 Patent's Specification

43. The '442 patent describes a “cluster computing system 10” comprising “a number of computation nodes 20 and a number of booster nodes 22” that are “arranged to compute a computation task ... comprising a plurality of sub-tasks.” EX1001, 1:66-2:5, 2:66-3:2. A computation node includes a “multi-core processor,” while a booster node includes an “accelerator-type processor.” *Id.*, 3:25-27, 3:33-36. The '442 patent explains that these devices and architecture were known in the art, with booster nodes known and used for processing “highly scalable code parts” or sub-tasks of a computation task. *Id.*, 1:23-55, 3:25-38, 3:53-55.



EX1001, FIG. 1 (annotated). The cluster includes a resource manager and an application manager, as well as daemons associated with each element of the cluster where each daemon “runs as a background process” in the respective element. EX1001, 3:6-23. The application manager collects information from daemons at the nodes and provides this information to the resource manager “to determine if the distribution of sub-tasks between computation nodes and booster nodes can be adapted or improved upon for a further computing iteration.” EX1001, 2:13-25. Although the resource manager and application manager are depicted as separate components, a computer scientist would not be concerned if it were one or more, or if one was a subcomponent of the other, because that is just a software structuring choice and does not bear at all on functionality.

44. According to the '442 patent, “[a] job to be computed by the system may comprise a number of tasks some of which or all may be repeated a number of times during the execution of the job.” EX1001, 3:44-49. The '442 patent briefly describes distributing sub-tasks between computation nodes and boosters, but that an initial distribution of sub-tasks between a computation node and a booster during a first iteration may not be optimal, so altering the distribution of the sub-tasks among nodes for later iterations “might improve the efficiency of the computation of the task.” EX1001, 3:59-4:3.

B. The Prosecution History

45. During the prosecution of the '442 patent, the Patent Office Examiner initially rejected all claims as anticipated (fully disclosed) by Lippert (EX1004). EX1002, 155-166. In response, the Applicant argued that Lippert does not anticipate the claims because Lippert does not disclose processing the plurality of sub-tasks in “multiple computing iterations.” *Id.*, 174 (emphasis omitted). Applicant further argued that the claims were distinguished from Lippert because they “allow for redistribution based on information relating to the processing of the plurality of sub-tasks,” whereas according to Applicant, “Lippert merely proposes a load balancing based on the workload of the nodes.” *Id.*, 174-175.

46. The Examiner then allowed all pending claims. *Id.*, 184.

47. In my view, the Examiner should not have been persuaded by the Applicant's arguments, in part because iterative processing has been a basic feature of computing systems since their inception.

IX. CLAIM CONSTRUCTION

48. For this declaration, I apply the plain and ordinary meaning of all claim terms, and do not rely on any specific constructions. I reserve the right to offer my opinions on any constructions or construction issues presented by Patent Owner ParTec.

**X. CLAIMS 1-10 ARE OBVIOUS
IN VIEW OF LIPPERT AND BUDENSKE**

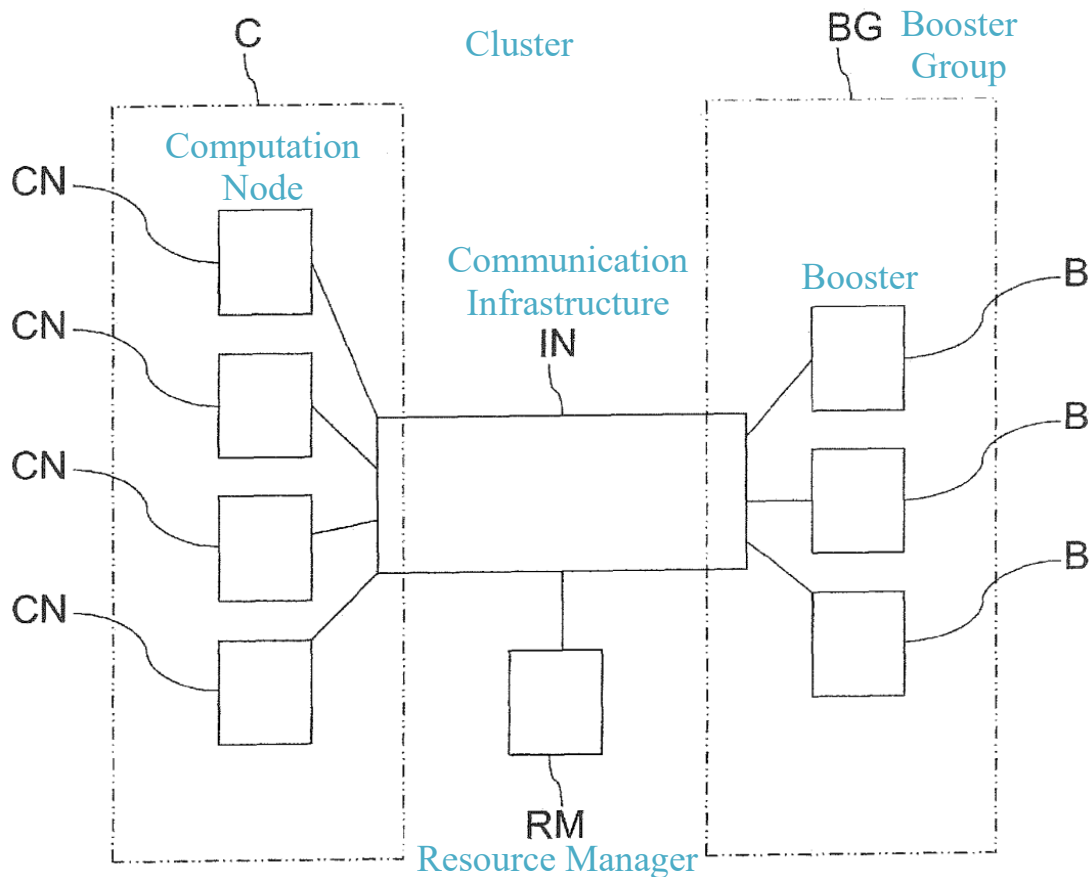
49. As explained below, claims 1-10 are obvious over Lippert in view of Budenske.

A. Lippert (EX1004)

50. Lippert is a U.S. patent application published on October 24, 2013. EX1004, Cover. Lippert is the same Thomas Lippert named as the first co-inventor on the '442 patent.

51. Lippert describes a "computer cluster arrangement for processing a computation task," the arrangement comprising a plurality of computation nodes and a plurality of boosters. EX1004, [0009]-[0011], FIG. 2. In Figure 2, Lippert depicts an example computer cluster with a plurality of computation nodes CN and a

plurality of boosters B in a booster group BG; the computation nodes CN and the boosters B are interconnected via a communication infrastructure IN. EX1004, [0067]. To process a computation task, a resource manager RM “initializes a first assignment and further on establishes a dynamic assignment of boosters B to computation nodes CN.” EX1004, [0068], [0013].



EX1004, FIG. 2 (annotated).

52. The computation task is allocated among computation nodes and boosters in part based on the processing requirements of each part of the computation task and the processing capacities, where computation nodes compute the first part

of the computation task while a booster computes a second part of the computation task. EX1004, [0017], [0010]-[0012], [0014]-[0016], [0018]-[0021], [0027]-[0029].

B. Budenske (EX1005)

53. Budenske is a printed publication published October 1998 in Volume 12, Number 4 of the Journal of Supercomputing. EX1005, 387. The Journal of Supercomputing is a well-regarded journal in the art, and Budenske would have been readily available to a person interested in the art around the time of its publication date. I am particularly familiar with the Journal of Supercomputing, as I served as a reviewer for this journal in 2022. EX1003, Exhibit A.

54. Budenske teaches the execution of iterative tasks with a “heterogeneous parallel hardware platform,” and the real-time re-mapping of sub-tasks to processors between iterations based on the sub-tasks’ processing of input data in a preceding iteration. EX1005, 387. An “Intelligent Operating System” (IOS) “make[s] a heuristically-based decision during the execution of the application whether to perform a remapping based on information generated by the application from its input data.” *Id.*, 387-388. A central module of the IOS (HC Kernel or HC Kernel Monitor) (1) establishes an initial mapping of the sub-tasks onto the processing resources of the heterogeneous computing system, (2) monitors the execution of the sub-tasks, and (3) at the end of each iteration, decides whether and how to change the mapping of the sub-tasks onto the hardware based on information about dynamic

parameters obtained from processing the sub-tasks in the preceding iteration.
EX1005, 401-402.

55. Budenske teaches determining a new mapping of sub-tasks to processing resources “in real time after all subtasks’ implementations for the current iteration have finished executing and before any subtask implementations begin to execute for the next iteration.” EX1005, 402. Budenske teaches “reconfigur[ing] the assignment of processing resources to subtasks **to reduce execution time** of the next iteration.” EX1005, 391 (emphasis added).

C. Combining Lippert And Budenske

56. A POSITA would have been motivated, with a reasonable expectation of success, to combine Lippert’s teachings on a computer cluster arrangement with dynamic assignments of booster nodes to computation nodes with Budenske’s teachings on re-mapping sub-tasks between iterations of an iterative application. A skilled artisan would have combined these respective teachings into an improved design (hereinafter “Lippert-Budenske”) that included benefits from each reference.

57. The resulting combination has the following teachings and elements cooperating as follows:

- A computer cluster, as taught by Lippert, including:
 - A plurality of computation nodes;
 - A plurality of booster nodes; and

- A communication infrastructure interfacing each of the computation nodes and the booster nodes;
- A resource manager module implemented as a daemon, as taught by Lippert (*e.g.*, EX1004, [0014]), arranged to:
 - Assign booster nodes to computation nodes for processing a computation task, as taught by Lippert (*e.g.*, EX1004, [0014]);
 - Assign, according to an initial mapping, sub-tasks of the computation task to the nodes for processing in a first iteration of the computation task, as taught by Budenske (*e.g.*, EX1005, 401-402); and
 - Dynamically assign, during the processing of the computation task, computation nodes, and booster nodes to each other, as taught by Lippert (*e.g.*, EX1004, [0013]);
- An application manager module, as taught by Budenske (*e.g.*, EX1005, 388, 402), arranged to:
 - Deploy the sub-tasks to the computation nodes and the booster nodes, as assigned for a current iteration, for execution of the sub-tasks by the nodes in the current iteration, as taught by Budenske (*e.g.*, EX1005, 402);

- Monitor execution of sub-tasks to collect information relating to the processing of the sub-tasks in the current iteration, as taught by Budenske (*e.g.*, EX1005, 402); and
- Assign, according to a new mapping obtained using information related to the processing of the sub-tasks, the sub-tasks to the nodes for a subsequent iteration of the computation task, as taught by Budenske (*e.g.*, EX1005, 401-402).

58. A POSITA would have been motivated, with a reasonable expectation of success, to so combine the teachings of Lippert and Budenske for at least the following reasons.

59. *First*, Lippert and Budenske each disclose heterogeneous computing systems that process an application or computation task with dynamic assignments. *E.g.*, EX1004, [0015]; EX1005, 387. Lippert discloses a flexible assignment of boosters to computation nodes to facilitate scalability, where parts of a computation task are allocated between computation nodes and boosters. *E.g.*, EX1004, [0013], [0015]-[0020]. Budenske discloses a flexible assignment of sub-tasks of an iterative application to processors of a heterogeneous computing system so that each sub-task is assigned “where it is best suited for execution” for each iteration. EX1005, 390, 387. A POSITA would have been motivated to implement the heterogeneous computing system of Lippert with the dynamic, iterative sub-task assignment

method of Budenske partly because doing so would facilitate efficient distributions of sub-tasks and processing resources during execution of an iterative application, thus fully leveraging the flexibility of the heterogeneous architecture and “reduc[ing] execution time” of computing iterations. EX1005, 391.

60. *Second*, Budenske provides express motivation for implementing its method in a heterogeneous computing system such as Lippert, namely to “fully utilize the architectural flexibility of such a [heterogeneous parallel computing] system.” EX1005, 390. A POSITA would have found that the dynamic flexibility of the heterogeneous computing system taught by Lippert through the “loose coupling” of computation nodes and boosters (EX1004, [0013], [0015] (describing “a maximum flexibility in providing hardware resources”)), naturally complements the dynamic flexibility of remapping sub-tasks among processing resources to account for the actual processing of the sub-tasks by the processing resources during preceding iterations, as taught by Budenske (EX1005, 391), and that the combination of these teachings would provide increased flexibility for optimally processing iterative applications.

61. *Third*, Lippert recognizes that it has long been known to outsource portions of computation tasks “of high resource requirements” to accelerators (boosters), while other portions of the computation tasks are executed by computation nodes. EX1004, [0003], [0028]-[0029] (describing how computation

nodes and boosters are “typically” configured with processors suited for different tasks), [0068], [0084]-[0086]. Budenske similarly recognizes the varying requirements of different computations of a given computation task, and that “[h]eterogeneous parallel architectures are ideal computing platforms for efficiently handling computational tasks with such diverse requirements.” EX1005, 390 (discussing a computation task that involves “numeric computation, quasi-symbolic computation ..., and symbolic computation”). A POSITA would have recognized that Lippert and Budenske follow the same principles of efficient processing of a task, comprising a plurality of sub-tasks, with a heterogeneous computing system, by assigning sub-tasks to processors (e.g., computation nodes or boosters) best suited to execute the sub-task. A POSITA thus would have been motivated to implement Budenske’s teachings with a heterogeneous computing system including boosters, as taught by Lippert, partly because “the boosters are implemented to process specific problems at high speed.” EX1004, [0028].

62. *Fourth*, a POSITA reading Lippert would have understood the need to account for sub-task assignment among nodes for iterative applications, which Budenske teaches. While Lippert focuses on the configuration of the processing resources of the system itself, Lippert only broadly and generally discusses the nature of the “application” or “computation task” being computed. *E.g.*, EX1004, [0015]-[0016]. Further, Lippert discloses dynamically assigning boosters to

computation nodes during runtime, and outsourcing parts (i.e., sub-tasks) of the computation tasks to the assigned boosters, suggesting that sub-tasks are also dynamically remapped to newly assigned boosters during processing of the task and “in dependency of processing a specific computation task.” EX1004, [0013] (“establish a dynamic assignment at runtime, which means during processing of the computation task”), [0015] (“booster allocation is performed as a function of application needs, which means in dependency of processing a specific computation task”). Budenske, meanwhile, focuses on applications where sub-tasks are iteratively and concurrently executed, and where the processing requirements for a sub-task may evolve over time. EX1005, 387, 390. To that end, Budenske teaches monitoring dynamic parameters of the application during each iteration—where these dynamic parameters relate to the processing of the input data and “have the most impact on the execution time of the subtasks in the application”—and adjusting the assignment of sub-tasks for the next iteration accordingly based on the dynamic parameters. EX1005, 391, 393. A POSITA thus would have naturally turned to Budenske when implementing Lippert for iterative applications in order to “assign each ... subtask to the processors where it is best suited for execution” and thus minimize execution time as the processing requirements of the computation task evolves over time. EX1005, 390.

63. *Fifth*, a POSITA would have recognized the compatibility of Lippert and Budenske's teachings. Lippert, for example, recognizes that a computation task can be divided into at least a first part for computing by computation nodes, and a second part for computation by boosters. EX1004, [0009]-[0014], [0016]-[0020], [0068]. Further, code blocks (e.g., sub-tasks) are classified as scalable or complex. EX1004, [0084]. A POSITA would have understood that the "first part of the computation task" computed by the computation nodes includes the less scalable, complex sub-tasks, while the "second part of the computation task" computed by the booster(s) includes the highly scalable sub-tasks. *Id.*; EX1014, 5. Further, Lippert recognizes that "[e]ach code has highly scalable and less scalable complex elements," and so a POSITA would have been motivated to change which sub-tasks are assigned to a first part (computed by computation nodes) and a second part (computed by boosters), regardless of a sub-task's initial classification as complex or scalable (and thus initial mapping to computation nodes or boosters), if information relating to the processing of the sub-task in a preceding iteration suggests that a different node would better process the sub-task.

64. Nothing in these references teaches away from this combination. On the contrary, a POSITA would have recognized that the heterogeneous computing systems of Budenske and Lippert are similar. Specifically, Budenske discloses a heterogeneous computing system including DSP (digital signal processors) and

RISC (reduced instruction set computer) processors (EX1005, 391), and a POSITA would have recognized that the DSPs are processors that function as accelerators/boosters and the RISC processors as computing nodes, much like the processing elements in Lippert. A POSITA thus would have understood that the teachings of Lippert and Budenske are compatible and would work well together.

65. *Finally*, a POSITA would have reasonably expected to succeed in implementing the Lippert-Budenske combination because doing so would only entail providing software code, which was well within the skill of a POSITA. Further, a POSITA would have reasonably expected success implementing the teachings of Budenske in a heterogeneous computing system as taught by Lippert partly because there is nothing incompatible between the two approaches, and because Budenske explains that its teachings “can also be used for other application domains and classes of hardware platforms whose characteristics are similar to those of the application and platforms considered here.” EX1005, 404. Other prior art references available to the POSITA disclosed successful implementations of Budenske’s teachings and confirmed that Budenske’s approach to re-mapping sub-tasks is feasible and achieves faster execution times compared to other dynamic task scheduling techniques. EX1007, 78-79 (describing Budenske), 92 (illustrating execution times of Budenske’s “On-Off” approach compared to other dynamic scheduling approaches for task graphs of various sizes and types); EX1008, 156-165

(discussing Budenske's approach and performance results). In view of these references, a POSITA would have reasonably expected to succeed implementing Lippert-Budenske and achieving the benefits described in each reference.

66. Further, the '442 patent provides only a high-level description of adjusting the distribution of sub-tasks between computation nodes and boosters. *E.g.*, EX1001, 3:59-4:3, 4:23-31. This level of disclosure reflects the state of the art, as it assumes anyone desiring to carry out the process would know of the equipment and techniques to be used, none being specifically described, and thus confirms that POSITAs would have reasonably expected to succeed in implementing Lippert-Budenske.

67. So combined, the Lippert-Budenske combination dynamically distributes sub-tasks of a computation task among computation nodes and boosters of a heterogeneous computation system between iterations of the computation task, to more efficiently process the computation task, and thus renders all challenged claims obvious.

1. **Claim 1**

a) **Element [1.1]**²

[1.1] A method of operating a heterogeneous computing system comprising a plurality of computation nodes and a plurality of booster nodes, at least one of the plurality of computation nodes and a plurality of booster nodes being arranged to compute a computation task, the computation task comprising a plurality of sub-tasks, the method comprising:

68. Lippert-Budenske discloses and teaches this element. Lippert and Budenske each disclose a method of operating a heterogeneous computing system to compute a computation task. Lippert discloses operating a heterogeneous computing system comprising a plurality of computation nodes and a plurality of booster nodes, with at least one of the nodes being arranged to compute a computation task comprising a plurality of sub-tasks. Specifically, Lippert discloses a “computer cluster arrangement for processing a computation task,” where the “computer cluster arrangement” comprises a plurality of computation nodes and a plurality of boosters (booster nodes). EX1004, [0009]-[0013]. Processors applied in computation nodes are different from the processors applied in boosters. EX1004, [0029]. A POSITA thus would have understood that the computer cluster arrangement is a heterogeneous computing system because the computer cluster

² Reference numbers in the format of [claim#.element#] are added throughout for ease of reference.

arrangement comprises different types of processors, arranged as computation nodes and booster nodes. Figure 2 of Lippert, reproduced below, “shows a computer cluster arrangement comprising a cluster C as well as a booster group BG,” where the cluster as depicted comprises “four computation nodes, also referred to as CN, as well as three boosters, also referred to as B.” EX1004, [0067]. While Figure 2 labels the computation nodes as the cluster “C,” a POSITA would understand that the entire arrangement depicted in Figure 2 is the cluster, as stated in Lippert: “FIG. 2 shows a schematic illustration of a computer cluster arrangement.” EX1004, [0059].

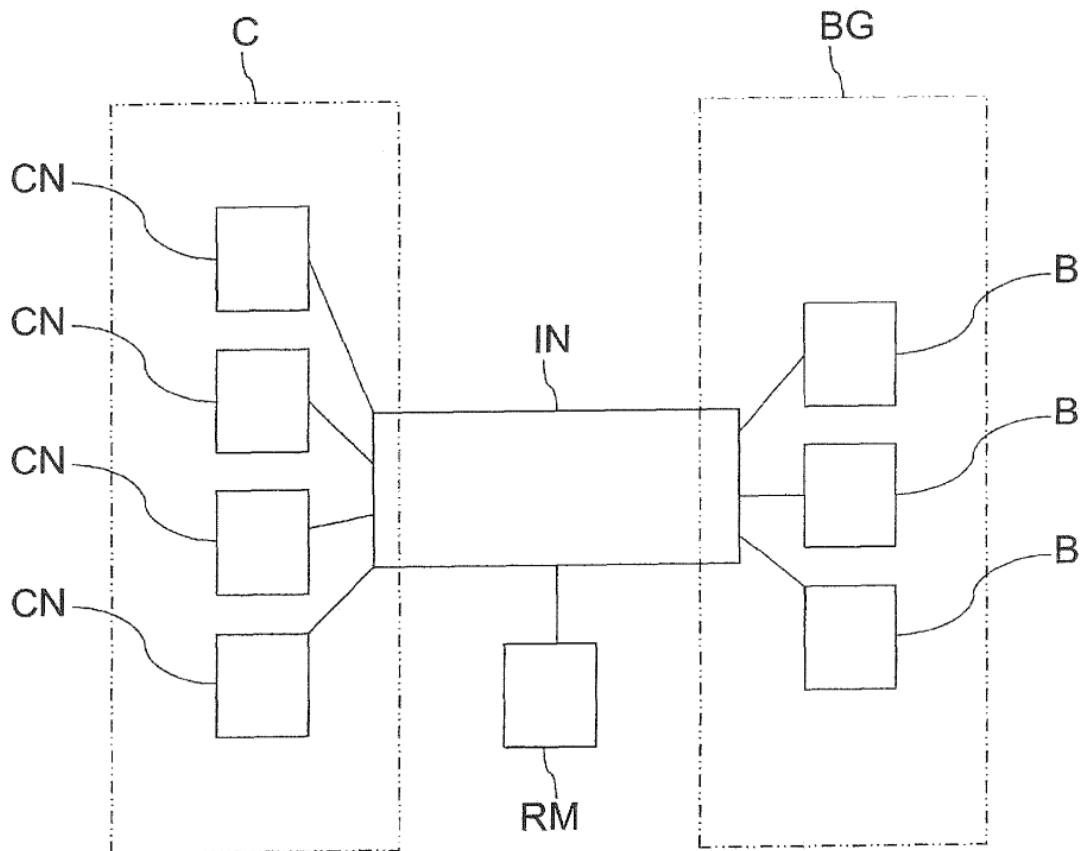


Fig.: 2

EX1004, FIG. 2.

69. Lippert discloses that “the computation task may comprise several sub[-]problems, also referred to as sub[-]tasks, which in their entirety describe the overall computation task.” EX1004, [0016]. The computation task may be divided “into several parts,” and the computer cluster arrangement is arranged “to solve the parts of the computation task in parallel or in succession.” EX1004, [0016]. Thus, Lippert (and Lippert-Budenske) discloses this claim element.

70. Budenske discloses a method for operating a heterogeneous computing system to compute a computation task, the computation task comprising a plurality of sub-tasks. Budenske discloses a “methodology for real-time on-line input-data dependent remapping of the application subtasks to the processors in the heterogeneous parallel hardware platform.” EX1005, 387. Budenske explains that the configuration of the heterogeneous parallel hardware platform (heterogeneous computing system) depends on the expected needs of the application (task) to be computed, and as an example, teaches a heterogeneous parallel hardware platform that includes “up to four different types of processors, and up to a total of 64 processors (of all types combined).” EX1005, 391.

71. As explained by Budenske, an “application ... can be modeled as an iterative execution of a set of partially ordered subtasks.” EX1005, 390. An “application task” is represented as “a *DDG* (data dependency graph), whose nodes

are the subtasks that need to be executed to perform the application and whose arcs are the data dependencies among subtasks.” EX1005, 391. The DDG is “structured as a directed acyclic graph (*DAG*).” EX1005, 391.

72. In combination, Lippert-Budeske discloses a method of operating a heterogeneous computing system comprising a plurality of computation nodes and a plurality of booster nodes, at least one of the plurality of computation nodes and a plurality of booster nodes being arranged to compute a computation task, the computation task comprising a plurality of sub-tasks.

b) Element [1.2]

[1.2] in a first computing iteration, assigning and processing the plurality of sub-tasks by at least a portion of the plurality of computation nodes and at least a portion of the plurality of booster nodes in a first distribution; and

73. Lippert-Budenske discloses and teaches this element. Lippert discloses computing a “first part of the computation task” and a “second part of the computation task” (*the plurality of sub-tasks*) by, respectively, “at least two of the plurality of computation nodes CN” (*at least a portion of the plurality of computation nodes*) and “at least one booster B” (*at least a portion of the plurality of booster nodes*). EX1004, [0091]. A POSITA would have understood this computing of the parts of the computation task by at least two of the plurality of computation nodes and at least one booster as *assigning and processing* the parts of the computation task *in a first distribution*, at least because Lippert teaches

“establishing a static assignment [of a booster to a computation node] at start of a processing of a computation task,” where “assignment information [is provided] to the computation nodes for outsourcing parts of the computation tasks from at least one computation node to at least one booster.” EX1004, [0013]-[0014]. A POSITA would have understood “distribution” as used here in the claim to refer to how the sub-tasks are assigned to nodes (and thus distributed among the nodes), as opposed to meaning a probability distribution like a Gaussian.

74. Although Lippert discloses the computation node “outsourcing” parts of the computation task to a booster where the computation node processes the first part in a first step and the booster node processes the second part in a second step (e.g., EX1004, [0091]-[0092]), suggesting that the parts are solved in succession, Lippert also discloses that the system may “solve the parts of the computation task *in parallel* or in succession.” EX1004, [0016] (emphasis added). Further, Lippert does not expressly specifically disclose the iterative execution of a computation task, and thus does not expressly specifically disclose a “first computing iteration,” but Lippert explains that “the person skilled in the art appreciates that any of the steps can be performed iteratively, in a different order and may comprise further sub steps.” EX1004, [0093].

75. Budenske discloses assigning a plurality of sub-tasks to processing resources of a heterogeneous computing system for processing in a first iteration.

For example, Budenske discloses, “[f]or the initial iteration through the set of subtasks” (*in a first computing iteration*), “assign[ing] resources (e.g., processors) to the subtasks.” EX1005, 391; *see also id.*, 396-97 (describing how to assign subtasks to processors). This “initial mapping” (*first distribution*) of the subtasks (*the plurality of sub-tasks*) to the processing resources is used for a first computing iteration of the application (*computation task*). EX1005, 395.

76. A POSITA would have implemented Lippert-Budenske to, in a first computing iteration of an iterative application, assign the plurality of sub-tasks to at least a portion of the computation nodes and at least a portion of the boosters according to an initial mapping, where the initial mapping maps sub-tasks to processors (e.g., computation nodes or boosters), and where their assigned computation nodes and boosters thus process the plurality of sub-tasks. The POSITA would have been motivated to implement Lippert-Budenske in this way at least because doing so would ensure that sub-tasks are assigned to processing resources “best suited” to compute the algorithms of the sub-tasks while accounting for the initial architecture of computation nodes and boosters, and characteristics of the application itself (e.g., what algorithms will be used to compute each sub-task; what are the data dependencies between sub-tasks; etc.). *E.g.*, EX1005, 392, 396-397. Lippert-Budenske thus teaches and renders obvious, in a first computing iteration, assigning and processing the plurality of sub-tasks by at least a portion of the

plurality of computation nodes and at least a portion of the plurality of booster nodes in a first distribution.

c) **Element [1.3]**

[1.3] generating, using information relating to the processing of the plurality of sub-tasks by at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes, a further distribution of the plurality of sub-tasks between the plurality of computation nodes and the plurality of booster nodes for processing thereby in a further computing iteration.

77. Lippert-Budenske discloses and teaches this element. After processing the sub-tasks with the “initial mapping,” Budenske teaches “monitor[ing] the run-time values of the dynamic parameters at the end of each iteration through the underlying DDG to make a heuristically-based decision whether to continue with the current mapping or to select and instantiate a new mapping (for the next iteration).” EX1005, 395. Budenske thus discloses using values of dynamic parameters (*information relating to the processing of the sub-tasks by at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes*) to generate a “new assignment” (*further distribution*) of sub-tasks to resources “to use for the next execution iteration through the subtasks” (*in a further computing iteration*). EX1005, 391.

78. Specifically, “[a]fter each execution iteration through the set of subtasks, the values of certain dynamic parameters of the application may change, such as the number of objects detected in the current frame of a real-time image

stream being processed.” EX1005, 391. The dynamic parameters are “computed by the application as it executes.” EX1005, 393. The values for the dynamic parameters are thus obtained after processing the sub-tasks in an iteration: “After all subtasks have completed execution for a given iteration through the DDG, and before the next iteration begins, the latest values of these dynamic parameters will be reported to the on-line HC Kernel.” EX1005, 391. At each iteration, there can be a different mapping of the DDG (parallelism) onto the accelerators. The latest values of dynamic parameters are an example of the claim’s “information relating to the processing of the plurality of sub-tasks.” The “most recent values of such dynamic parameters” are then used to determine whether “to reconfigure the assignment of processing resources to subtasks to reduce execution time of the next iteration,” and “[i]f it is desirable, the HC Kernel will select a new assignment to use for the next execution iteration through the subtasks.” EX1005, 391. In particular, Budenske teaches generating a “vector of D representative values” as an “approximation of the set of current actual dynamic parameters,” and using that vector to identify the optimal mapping of sub-tasks from the possible mappings (tracked in a “Mapped DDG” or “MDDG Table”). EX1005, 402. Budenske thus teaches selecting a new mapping from a set of candidate mappings that best matches the values of the dynamic parameters (*the information*), and instantiating (*generating*) the further distribution according to the selected mapping for the next computing iteration.

EX1005, 402-403. Budenske also discloses an alternative wherein a fully “on-line heuristic” is used, such that the new mapping is generated in real-time between iterations based on the information. EX1005, 389-90, 398, 403.

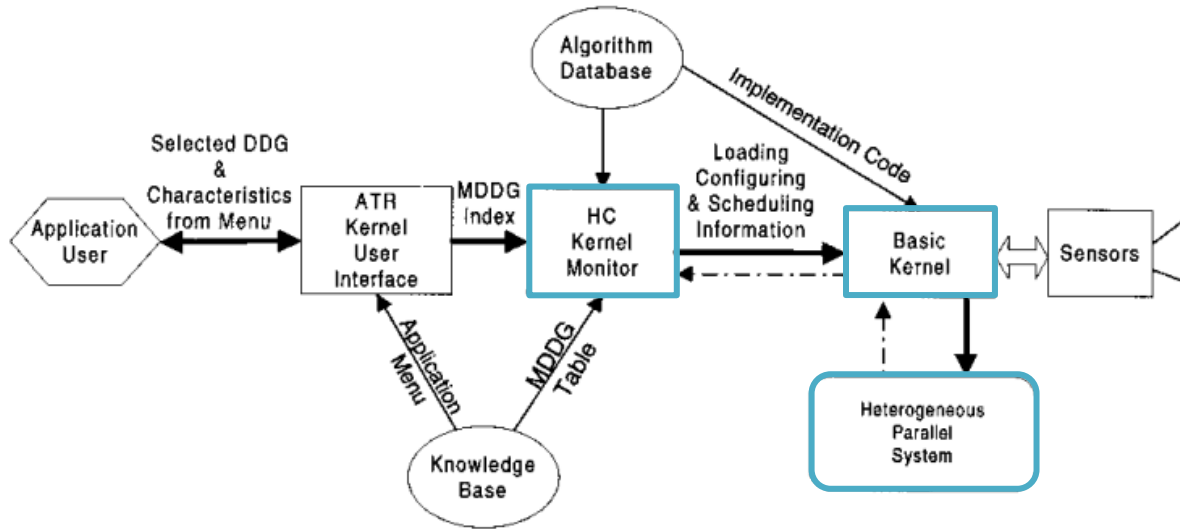
79. A POSITA would have found it obvious to implement the heterogeneous computing system of Lippert with the dynamic sub-task mapping method of Budenske such that, between iterations, Lippert-Budenske uses values of dynamic parameters calculated during a preceding iteration to generate a new assignment of sub-tasks to the plurality of computation nodes and the plurality of boosters for processing in the next iteration. The POSITA would have been motivated to implement Lippert-Budenske in this way partly in order “to reduce execution time of the next iteration.” EX1005, 391. The POSITA would have further been motivated to implement Lippert-Budenske in this way because Budenske explains that, “[b]ecause the execution time of application subtasks ... is highly input-data dependent ..., this matching and scheduling of application subtasks to processors *must* be performed dynamically at run time.” EX1005, 390 (emphasis added). The POSITA would have reasonably expected to succeed in implementing Lippert-Budenske partly because Lippert and Budenske each provide ample implementation details for achieving the disclosed functionality, and a POSITA would have found it simple and straightforward to combine the teachings. Lippert-Budenske thus teaches and renders this element and this claimed method obvious.

2. Claim 2

2. The method according to claim 1, wherein an application manager receives the information and determines the further distribution.

80. Lippert-Budenske discloses and teaches this element, and renders this claim obvious for the foregoing and following reasons. Lippert-Budenske renders claim 1 obvious, as discussed above in Section X.C.1. Budenske discloses an “HC Kernel” (also referred to as “HC Kernel Monitor”) and a “Basic Kernel” that cooperate to manage an application, and thus a POSITA would have understood aspects of the HC Kernel and the Basic Kernel together as an *application manager*. EX1005, 388-89, 401-02. According to Budenske, “the HC Kernel interacts with the Basic Kernel ... to execute the application and monitor its execution so it can decide when to remap the subtasks onto the hardware platform” (EX1005, 388), where the Basic Kernel receives “the latest values of ... dynamic parameters” at the end of each iteration (*receives the information*). EX1005, 402 (describing how the Basic Kernel “begin[s] execution of the application task”). The Basic Kernel sends the latest values of the dynamic parameters to an HC Kernel Monitor, which then “use[s] the most recent values of these dynamic parameters to estimate if changing the matching and scheduling will reduce the expected execution time of the next iteration through the corresponding DDG.” EX1005, 402. Figure 3 of Budenske depicts the Basic Kernel and the HC Kernel Monitor (together, *the application*

manager) interacting with the heterogeneous parallel computing system during execution of an iterative application:



EX1005, Figure 3 (annotated); *see also id.*, Figure 1.

81. A POSITA would have understood the HC Kernel and the Basic Kernel of Budenske together as an “application manager” that “receives the information and determines the further distribution” because the Basic Kernel receives the values of the dynamic parameters (*receives the information*) from the heterogeneous parallel computing system and inputs the dynamic parameter values to the HC Kernel to determine a new assignment of sub-tasks to processing resources (*determines the further distribution*). EX1005, 391, 402. Lippert-Budenske thus discloses and teaches this claim element and renders this claim obvious.

3. Claim 3

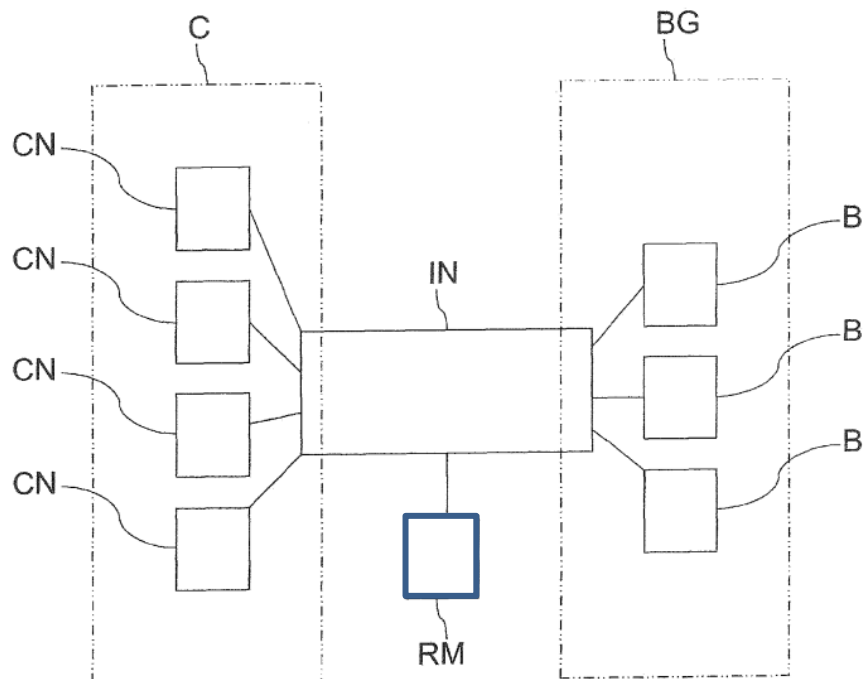
a) Element [3.1]

[3.1] The method according to claim 2, wherein a resource manager determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes for the first computing iteration as a function of the computation task and

82. Lippert-Budenske discloses and teaches this claim element and renders this claim obvious for the foregoing and the following reasons. Specifically, as discussed below, Lippert discloses a “resource manager,” and Budenske discloses an HC Kernel Monitor, and a POSITA implementing Lippert-Budenske would combine aspects of Lippert’s resource manager and Budenske’s HC Kernel Monitor into a combined resource manager that determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes for the first computing iteration as a function of the computation task. Lippert-Budenske thus includes a resource manager that manages resources of the heterogeneous computing system (e.g., computation nodes, booster nodes, and the assignment thereof for processing a computation task), and an application manager that manages applications (e.g., computation tasks) processed by the heterogeneous computation system, where the resource manager handles an initial mapping of sub-tasks because it manages the initial arrangement of the resources and does not depend on information from processing the sub-tasks. In contrast, the application manager

handles re-mapping of the sub-tasks to the resources because it receives information relating to the processing of the sub-tasks during an application's execution. *See* ¶¶80-81 (discussing the application manager of Lippert-Budenske); ¶¶83-85 (discussing the resource manager of Lippert-Budenske).

83. Lippert discloses a resource manager that determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes as a function of the computation task. The resource manager “establish[es] a static assignment at start of a processing of a computation task,” and “establish[es] a dynamic assignment at runtime, which means during processing of the computation task.” EX1004, [0013]; *id.*, [0014], [0017]. FIG. 2 of Lippert depicts the resource manager RM of the heterogeneous computing system.



EX1004, FIG. 2 (annotated). This resource manager RM is similar to the resource manager RM of the '442 patent. *See* EX1001, FIG. 1 (element 28). The *resource manager* of Lippert determines the computation node-booster assignment based on “computation task requirements,” and thus *as a function of the computation task*. EX1004, [0017]. Specifically, “booster allocation is performed *as a function of application needs*, which means in dependency of processing a specific computation task.” EX1004, [0015] (emphasis added). The resource manager “determin[es] an initial assignment of boosters to computation nodes” based on a predetermined assignment metric, and dynamically assigns boosters to computation nodes “at runtime” as the computation task requirements evolve and the assignment metric is accordingly altered. EX1004, [0017]; *id.*, [0013], [0030]. The assignment metric is managed by the resource manager. It is updated “as a function of a load balancing, which detects workload of the computer cluster arrangement, especially of the boosters.” *Id.*, [0017]. For the assignment metric, Lippert teaches “detect[ing] computing capacities of boosters and further detect[ing] computation task requirements,” so that a booster assigned to a computation node “provides the required capacities to the computation node.” *Id.* A POSITA would have understood the resource manager’s assignment based on “computation task requirements” as being determined *as a function of the computation task*. EX1004, [0017].

84. Budenske, meanwhile, teaches an HC Kernel Monitor that determines the assignment of the plurality of sub-tasks to the plurality of nodes for the first computing iteration as a function of the computation task. Specifically, Budenske teaches an HC Kernel Monitor that “establish[es] the initial mapping” of sub-tasks to processing resources for an initial iteration (*the assignment of the plurality of sub-tasks to the plurality of computation nodes ... for the first computing iteration*). EX1005, 402. The initial mapping is a mapped DDG (MDDG) for a given initial scenario (i.e., set of dynamic parameter values), where the MDDG is structured as a directed acyclic graph that arranges sub-tasks according to “data dependencies among subtasks” with algorithms selected for implementing the sub-tasks (*as a function of the computation task*). *Id.*, 391, 396, 402. A POSITA would have understood the HC Kernel Monitor partly as a *resource manager* because it “decide[s] how to assign resources (e.g., processors) to the subtasks.” EX1005, 391.

85. In combination, Lippert-Budenske includes a resource manager that determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes, as taught by Lippert, for the first computing iteration, as taught by Budenske. The assignment is determined as a function of the computation task, as taught by both Lippert and Budenske, because doing so would “enforce data-dependency constraints among the subtasks” (EX1005, 397) and ensure that the sub-tasks of an iterative application are assigned

“to the processors where [they are] best suited for execution” (EX1005, 390). *See also* EX1004, [0015], [0017].

b) Element [3.2]

[3.2] wherein the application manager receives the information and processes the information as input to the resource manager such that the resource manager dynamically alters further distribution during the computing of the computation task.

86. Lippert-Budenske discloses and teaches this element, thus rendering the claimed method obvious for the foregoing and the following reasons. As discussed above in Section X.C.2, Lippert-Budenske includes an application manager that receives the information (*the application manager receives the information*) and determines the further distribution (*processes the information*). Lippert-Budenske further includes a resource manager that “establish[es] a dynamic assignment at runtime, which means during processing of the computation task” (*dynamically alters further distribution during the computing of the computation task*), as taught by Lippert. EX1004, [0013]; Section X.C.3.a).

87. A POSITA implementing Lippert-Budenske would have found it obvious to input the new mapping of sub-tasks by the application manager into the resource manager such that the resource manager dynamically alters the distribution of computation nodes and booster nodes during the computing of the computation task, because doing so would utilize the “maximum flexibility in providing hardware

resources” taught by Lippert. EX1004, [0015]; *see also* EX1005, 390 (teaching to “fully utilize the architectural flexibility”). That is, in addition to re-mapping the sub-tasks to processing elements of the system as taught by Budenske, Lippert-Budenske also dynamically assigns computation nodes and boosters to each other “during processing of the computation task” as taught by Lippert (EX1004, [0013]). By providing the new mapping for the next iteration to the resource manager, Lippert-Budenske would, for example, allocate additional boosters for processing a given sub-task or otherwise re-configure computation node-booster assignment to implement the new mapping and “reduce execution time of the next iteration” (EX1005, 391). EX1004, [0014]-[0017], [0068]. In this way, Lippert-Budenske “provide[s] fault tolerance in case of accelerator failures” and avoids “a lack of resources or ... an excessive supply of resources” when executing an application. EX1004, [0003].

88. A POSITA would have understood that “the resource manager dynamically alters further distribution” as recited in this claim element *encompasses* “the resource manager dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other,” because claim 4 depending from claim 3 specifies this, and further because the ’442 patent describes the resource manager adjusting booster assignments according to updated assignment

information from the application manager. EX1001, 4:27-31. Thus, in view of the above, Lippert-Budenske discloses and teaches this element and this claim.

4. Claim 4

4. The method according to claim 3, wherein the resource manager receives the information such that the resource manager dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other during the computation of the computation task.

89. Lippert-Budenske discloses and teaches this element, and renders this claim obvious for the foregoing and the following reasons. As discussed above in Section X.C.3, the resource manager of Lippert-Budenske receives the information and dynamically alters the assignment of the plurality of computation nodes and the plurality of booster nodes to each other during the computation of the computation task. EX1004, [0013], [0017].

5. Claim 5

5. The method according to claim 1, wherein daemons operate in the plurality of computation nodes and the plurality of booster nodes to generate the information.

90. Lippert-Budenske discloses and teaches this claim element, and renders this claim obvious for the foregoing reasons addressing claim 1 and for the following reasons. Lippert-Budenske renders claim 1 obvious. Section X.C.1. Lippert discloses “detect[ing] computing capacities of boosters” (EX1004, [0017]), and Budenske teaches values of dynamic parameters being reported to the Basic Kernel

during each iteration (*e.g.*, EX1005, 402). In order to send the dynamic parameter values taught by Budenske, as well as information regarding computing capacities as taught by Lippert, from each computation node and each booster node to a central manager to determine a re-mapping of sub-tasks, a POSITA thus would have implemented Lippert-Budenske with daemons at each computation node and each booster node configured to collect and send this information. POSITAs knew that a daemon is a background process that performs a specific task, and that daemons were commonly implemented at compute nodes to execute tasks at the compute nodes and report status information back to a central resource manager for managing cluster resources. *E.g.*, EX1013, 5-7 (describing a central controller daemon and daemons running at each compute node “to export control to SLURM”), 20 (describing a daemon at each node “for managing user jobs and monitoring system state,” where its “most common action ... is to report system state on request”); EX1021, 7:66-67 (defining a “daemon” as “a program that runs unattended to perform a standard service”), 8:2-3 (“Daemon processes generally provide services that must be available at all times.”), 2:54-3:18 (describing using daemons to facilitate communication in a “parallel processing environment” comprising “a plurality of nodes connected through a network to one another”). A POSITA thus would have been motivated to implement background processes (*i.e.*, daemons) operating at each node that communicate the relevant information in a status update

for dynamically remapping the sub-tasks among the nodes, to facilitate inter-element communication while providing communication authentication beneficially, and to enable independent control of task execution and reporting of task processing at each node. *E.g.*, EX1012, 9 (describing a central daemon and node daemons or “TaskTrackers” for executing tasks at the node), 38-40 (describing a “NodeManager” daemon at each node “responsible for launching the applications’ containers, monitoring their resource usage ..., and reporting the same to the ResourceManager”); EX1013, 10 (describing how daemons authenticate and secure communications between nodes); EX1022, 3:63-66 (“As is commonly known, a daemon is a process that runs in the background and performs a specified operation at predefined times or in response to certain events.”), 3:66-4:14 (explaining that “[t]ypical daemon processes ... perform administrative tasks” and are implemented in “any environment ... which is capable of implementing background processes and supports communications between the various Nodes of the Cluster(s)”).

6. Claim 6

6. The method according to claim 1, wherein the first distribution is determined based on a rating provided in source code for each sub-task in the plurality of sub-tasks.

91. Lippert-Budenske discloses and teaches this claim element, and renders this claim obvious for the above reasons associated with claim 1 and the following reasons. Lippert-Budenske renders claim 1 obvious. Section X.C.1. Lippert

discloses determining the first distribution based on a rating provided in the source code for each sub-task in the plurality of sub-tasks.

92. Specifically, regarding *a rating*, Lippert discloses priority, complexity, and scalability information (individually and together, “*a rating*”) for each sub-task because the sub-tasks comprise the priority information, complexity information, and scalability information. Specifically, Lippert discloses that “[s]pecific *computation tasks may comprise priority information*, which indicates how urgently this specific computation task has to be computed.” EX1004, [0020] (emphasis added). This priority indicates “how urgent a processing of a computation task, or at least a part of a computation task, is compared to other parts of computation tasks being originated from other computation nodes.” *Id.* This relates to the dependency between sub-tasks, where the processing of one sub-task depends on the processing of another sub-task. Regarding complexity information and scalability information, Lippert discloses applications or computation tasks where “[e]ach code has highly scalable and less scalable complex elements,” broken down into “exascale code blocks” and “complex code blocks,” respectively (i.e., scalable sub-tasks and less-scalable complex sub-tasks). EX1004, [0084]. As “the distinction between highly scalable and complex is made on the level of code blocks,” the code blocks (sub-tasks) include the complexity information and scalability information. *Id.* Each sub-

task of the computation task thus comprises priority information, complexity information, and scalability information (*a rating*).

93. The rating is *provided in the source code for each sub-task in the plurality of sub-tasks* because the computation task “comprises” the information and is “defined by means of ... a source code.” EX1004, [0016].

94. Regarding *the first distribution is determined based on a rating*, Lippert further discloses a “predetermined assignment metric” for assigning nodes to each other, with parts of the computation task allocated accordingly, being specified as a function of “complexity information, scalability information,” and “priority information” (*a rating*). EX1004, [0023]; *id.*, [0018], [0043]. This information is specific to each sub-task, and the assignment in Lippert, including the initial distribution of sub-tasks among the nodes, is based partly on this information. *E.g.*, *id.*, [0015] (“booster allocation is performed as a function of application needs, which means in dependency of processing a specific computation task”), [0017]-[0020] (describing using the assignment metric to assign nodes). Lippert teaches adapting the architecture of the computation nodes and the boosters of a computing cluster to account for the scalability of the code blocks. EX1004, [0084]-[0088].

95. A POSITA would have understood that Lippert thus discloses and teaches determining the first distribution of sub-tasks based on a rating provided in source code for each sub-task, including a rating based on scalability information

(e.g., indicating that the code block/sub-task is more scalable or less scalable) to ensure that the part of a computation task distributed to a booster is suitable for the resource capacities of the booster. This is consistent with the disclosure in the '442 patent of an operator estimating a scalability factor for each sub-task and distributing the sub-tasks accordingly. EX1001, 4:16-47. Further, the '442 patent confirms that POSITAs already knew how to annotate application code to indicate that a given task is “highly scalable” and thus suitable for execution by boosters. EX1001, 1:39-55 (citing EX1014); EX1014, 15-16 (describing how to designate portions of code as sub-tasks as highly scalable and suitable for execution on specified hardware devices, e.g., boosters).

96. A POSITA would have found it obvious when implementing Lippert-Budenske to determine the initial mapping of sub-tasks to computation nodes and boosters based on a rating provided in source code for each sub-task, where such rating comprises an indication of scalability, as taught by Lippert. The POSITA would have been motivated to do so to create optimal assignments of sub-tasks to processing resources (e.g., computation nodes or boosters) that account for the suitability of the processing resource for the corresponding scalability/complexity of the sub-task while minimizing the expected execution time. The POSITA would have reasonably expected to succeed in implementing Lippert-Budenske in this way because Lippert provides ample implementation details for achieving the disclosed

functionality, and a POSITA would have found it simple and straightforward to combine the teachings. Lippert-Budenske thus renders this claim obvious.

7. **Claim 7**

7. The method according to claim 1, wherein the information is used to provide a grouping of sub-tasks in at least one of the first computing iteration and the further computing iteration.

97. Lippert-Budenske discloses and teaches this claim element, and renders this claim obvious for the foregoing reasons associated with claim 1 and the following reasons. Lippert-Budenske renders claim 1 obvious. Section X.C.1. Regarding *a grouping of sub-tasks*, Lippert discloses “divid[ing] the computation task into several parts,” including a first part computed by computation node(s) and a second part computed by booster(s). EX1004, [0016]; *id.*, [0009]-[0012]. A POSITA would have understood these “parts” of the computation task as groupings of sub-tasks, e.g., a first grouping of sub-tasks for processing by computation nodes and a second grouping of sub-tasks for processing by boosters, because Lippert discloses that a computation task “comprise[s] several sub problems, also referred to as sub tasks.” EX1004, [0016].

98. As discussed above in Section X.C.1.c), Lippert-Budenske discloses and teaches using information relating to processing the plurality of sub-tasks to generate a further distribution of the sub-tasks for a further computing iteration. Given this disclosure and teaching, a POSITA implementing Lippert-Budenske

would have been motivated, with a reasonable expectation of success, to use the values of the dynamic parameters (*the information*) from a preceding iteration to provide, via the new mapping of sub-tasks, a first grouping of sub-tasks and a second grouping of sub-tasks for processing respectively with computation node(s) and booster(s) in the next iteration (*the further computing iteration*), at least because Lippert expressly discloses providing such groupings of sub-tasks. EX1004, [0016]; *id.*, [0009]-[0012].

8. Claim 8

8. The method according to claim 3, wherein a daemon operating at a node generates a measure of a loading of the node during processing of a sub-task of the plurality of sub-tasks.

99. Lippert-Budenske discloses and teaches this claim element, and renders this claim obvious for the reasons associated with claims 1-3 and the following reasons. Lippert-Budenske renders claim 3 obvious, and further renders the daemons operating at the plurality of computation nodes and booster nodes obvious. Sections X.C.3, X.C.5. Lippert discloses “detect[ing] workload of the computer cluster arrangement, especially of the boosters,” as well as “computing capacities.” EX1004, [0017]. For Lippert’s resource manager to detect such information, a POSITA would have understood that a background process (i.e., a daemon) implemented at each node to measure and report workload and computing capacity information (*generates a measure of a loading of the node during the processing of*

a sub-task of the plurality of sub-tasks) would be the most common way to generate and communicate such information, and the easiest to implement because daemons already exist for performing this function. *E.g.*, EX1012, 38 (describing a per-node daemon “monitoring their resource usage (CPU, memory, disk, network) and reporting the same to the ResourceManager”); EX1013, 6-7 (describing a “daemon running on each compute node”), 19-20 (explaining how node daemons provide “resource consumption information ... associated with a job”).

100. A POSITA thus would have been motivated, with a reasonable expectation of success, to implement Lippert-Budenske with a daemon operating at each computation node and booster node that generates a measure of the loading (workload) of nodes during processing of a sub-task of the plurality of sub-tasks, because doing so would allow the dynamic remapping of sub-tasks to the heterogeneous computing system, as taught by Budenske, to also account for “load balancing” as taught by Lippert. EX1004, [0017].

9. Claim 9

[9.1] A heterogeneous computing system comprising:

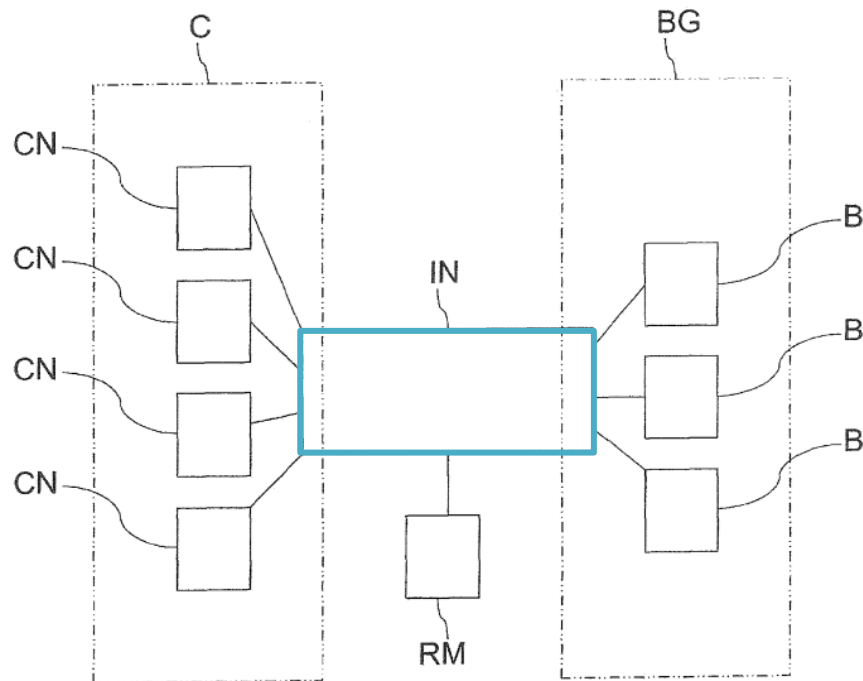
101. Lippert-Budenske discloses and teaches this element because Lippert and Budenske disclose a heterogeneous computing system. Section X.C.1.a).

[9.2] a plurality of computation nodes and a plurality of booster nodes for computing one or more tasks comprising multiple sub-tasks;

102. Lippert-Budenske discloses and teaches this element because Lippert discloses a plurality of computation nodes and a plurality of booster nodes for computing one or more tasks comprising multiple sub-tasks. Section X.C.1.a).

[9.3] a communication interface connecting the plurality of computation nodes with each other and the plurality of booster nodes;

103. Lippert-Budenske discloses and teaches this element because Lippert discloses the plurality of computation nodes and the plurality of booster nodes “interfacing a communication infrastructure.” EX1004, [0010]-[0011]; *id.*, [0016]. Thus, “the computation nodes as well as the boosters interact by means of the communication infrastructure.” *Id.*, [0016]. FIG. 1 of Lippert depicts the communication infrastructure IN, also called an interconnect, interfacing with each computation node CN and each booster B. *Id.*, [0067], FIG. 1.



EX1004, FIG. 1 (annotated).

104. Budenske similarly discloses that “[a]ll the processors of all types will have communication paths to one another,” via a “bus [that] can be used to provide communications among different collections of processors.” EX1005, 392.

105. Thus, when implementing Lippert-Budenske, a POSITA would have found it obvious for the heterogeneous computing system to include a communication interface connecting the plurality of computation nodes with each other and the plurality of booster nodes, at least because Lippert and Budenske each disclose such a communication interface.

[9.4] a resource manager for assigning at least a portion of the plurality of booster nodes and at least a portion of the plurality of computation nodes to each other for the computing of the one or more tasks in a first computing iteration; and

106. Lippert-Budenske discloses and teaches this element. Sections X.C.3, X.C.4.

[9.5] an application manager configured to receive information from daemons operating in at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes to update a distribution of the multiple sub-tasks between the plurality of computation nodes and the plurality of booster nodes in a further computing iteration.

107. Lippert-Budenske discloses and teaches this element. Section X.C.2. Lippert-Budenske thus discloses and teaches all elements and renders this claimed system obvious.

10. Claim 10

10. The heterogeneous computing system according to claim 9, wherein the resource manager receives the information such that the resource manager dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other.

108. Lippert-Budenske discloses and teaches this element, and thus renders this claim obvious. Sections X.C.9, X.C.4.

XI. CLAIMS 2-5 AND 8-10 ARE OBVIOUS IN VIEW OF LIPPERT, BUDENSKE, AND KAMBATLA

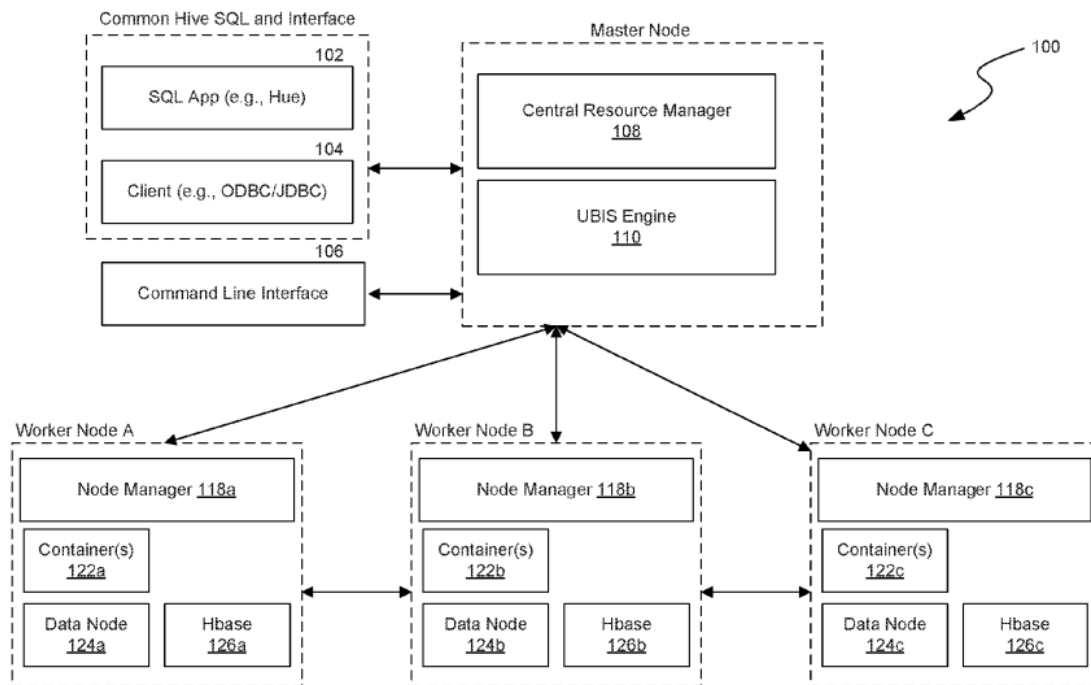
109. As explained below, claims 2-5 and 8-10 are obvious in view of Lippert, Budenske, and Kambatla. While Lippert and Budenske, in combination, render claims 1-10 obvious, the express teachings of Kambatla regarding resource management daemons reflect the knowledge of a POSITA and further confirm the obviousness of the claims.

A. Kambatla (EX1006)

110. Kambatla is a U.S. patent application filed May 15, 2017, and claiming priority to a provisional application filed September 14, 2016, and I understand this means that Kambatla qualifies as prior art under at least Section 102(a)(2). EX1006, Cover.

111. Kambatla discloses a resource management system for the “allocation and management of computing resources in distributed computing clusters.” EX1006, [0002]. Kambatla teaches techniques referred to as “utilization-based incremental scheduling” (UBIS) that “opportunistically allocate[s] computing resources not utilized by prior allocations.” EX1006, [0017]. Kambatla proposes UBIS as an improvement to known “cluster scheduler” frameworks such as Apache Hadoop YARN (“Yet Another Resource Negotiator”), used for scheduling (i.e., assigning) multiple tasks of a job to cluster computing resources. EX1006, [0004], [0020] (“UBIS can be implemented in or for use with Apache YARN”), [0023]-[0024], [0085].

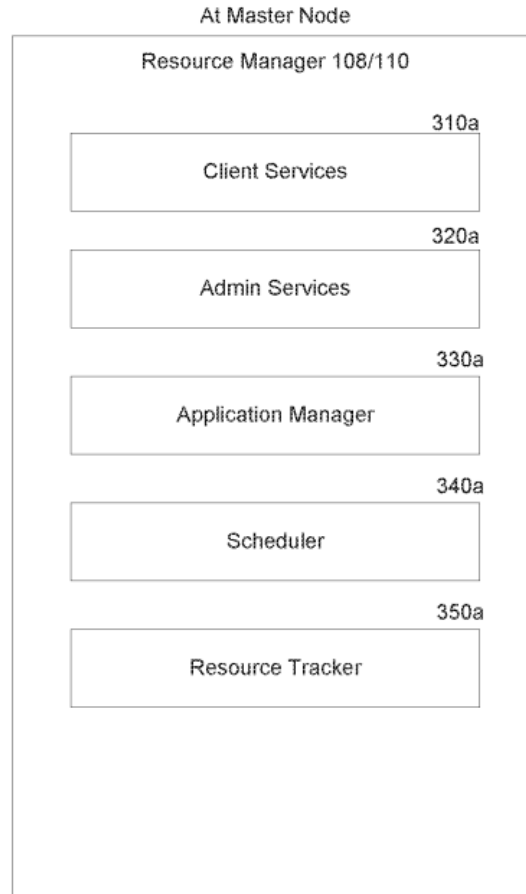
112. Kambatla discloses a computing environment, depicted in Figure 1, comprising a master node and a plurality of worker nodes in communication, where the master node assigns tasks of a job to the computing resources of the worker nodes for execution. EX1006, [0021]-[0022].



EX1006, FIG. 1. To facilitate allocating computing resources for tasks, Kambatla discloses daemons at each computing node, including a central resource manager daemon at the master node and a node manager daemon at each worker node. EX1006, [0030]-[0036] (describing resource manager), [0037]-[0041] (describing node managers), FIGS. 3A-B. The resource manager is “configured to manage and arbitrate resources among applications in the system.” EX1006, [0023].

113. Although Figure 1 depicts the resource manager 108 and the UBIS engine 110 as separate components of the master node, Figure 3A of Kambatla discloses a single resource manager comprising both components. EX1006, [0023], [0030], FIG. 3A. The resource manager comprises, among other things, “an

application manager 330a, a scheduler 340a, and a resource tracker 350a.” EX1006, [0031].



EX1006, FIG. 3A.

B. Combining Lippert, Budenske, And Kambatla

114. A POSITA would have found it obvious to implement a combination of Lippert and Budenske, as discussed in Section X.C. A POSITA further would have been motivated, with a reasonable expectation of success, to combine the teachings of Lippert-Budenske with Kambatla’s resource scheduling daemons. A skilled artisan would have combined these respective teachings into an improved

design (hereinafter “Lippert-Budenske-Kambatla”) that included benefits from each reference. The resulting combination has the following teachings and elements cooperating as follows:

- A computer cluster, as taught by Lippert, including:
 - A plurality of computation nodes;
 - A plurality of booster nodes; and
 - A communication infrastructure interfacing each of the computation nodes and the booster nodes;
- A resource manager module implemented as a daemon, as taught by Lippert (*e.g.*, EX1004, [0014]), arranged to:
 - Assign booster nodes to computation nodes for processing a computation task, as taught by Lippert (*e.g.*, EX1004, [0014]);
 - Assign, according to an initial mapping, sub-tasks of the computation task to the nodes for processing in a first iteration of the computation task, as taught by Budenske (*e.g.*, EX1005, 401); and
 - Dynamically assign, during the processing of the computation task, computation nodes and booster nodes to each other, as taught by Lippert (*e.g.*, EX1004, [0013]);
- An application manager module, as taught by Kambatla, arranged to:

- Deploy the sub-tasks to the computation nodes and the booster nodes, as assigned for a current iteration, for execution of the sub-tasks by the nodes in the current iteration, as taught by Budenske (*e.g.*, EX1005, 402);
- Monitor execution of sub-tasks to collect information relating to the processing of the sub-tasks in the current iteration, as taught by Budenske (*e.g.*, EX1005, 402); and
- Assign, according to a new mapping obtained using information related to the processing of the sub-tasks, the sub-tasks to the nodes for a subsequent iteration of the computation task, as taught by Budenske (*e.g.*, EX1005, 401-402);
- A node manager daemon at each computation node and booster node, as taught by Kambatla, arranged to:
 - Generate information relating to the processing of a sub-task at the node; and
 - Report the information to the application manager module.

115. A POSITA would have been motivated, with a reasonable expectation of success, to combine Lippert, Budenske, and Kambatla by implementing the computer cluster of Lippert with the dynamic re-mapping of sub-tasks between

iterations of an iterative application as taught by Budenske, and with the resource management daemons as taught by Kambatla, for at least the following reasons.

116. *First*, Kambatla, like Lippert-Budenske, discloses a heterogeneous computing system that processes a task comprising a plurality of sub-tasks. EX1006, [0004], [0021], [0059]. Kambatla further teaches that such systems enable the processing of not just one task or application but multiple tasks concurrently, with the system's processing resources being shared by the multiple tasks. It also provides techniques for optimally scheduling tasks across the processing resources. EX1006, [0004], [0017]. A POSITA thus would have been motivated to incorporate the complementary teachings of Kambatla into Lippert-Budenske to leverage better the full processing capacity of the heterogeneous computing system.

117. *Second*, a POSITA implementing Lippert-Budenske would have recognized the need to facilitate inter-element communication to allow the resource manager to allocate sub-tasks to nodes efficiently, and Kambatla teaches how to do so. For example, while Lippert discloses the computation nodes, booster nodes, and resource manager communicating with each other via the communication infrastructure (*e.g.*, EX1004, [0016]), Kambatla describes a node manager daemon operating at each node to communicate with a central resource manager daemon and to manage the execution of tasks at the node. EX1006, [0023], [0038]-[0041]. A POSITA thus would have naturally turned to the teachings of Kambatla for

implementation details on communicating information (i.e., values of dynamic parameters, processing capacities of the node) to the central resource manager.

118. *Third*, the particular teachings of Kambatla follow the same principles as Lippert-Budenske regarding the utilization of processing resources. For example, Kambatla describes the prior art background of allocating cluster computing resources to “jobs [that] typically include multiple tasks,” and the problem of “resource fragmentation and severe under-utilization of the computing resources in the cluster.” EX1006, [0003]-[0005]; *see also* EX1004, [0005] (discussing the problem of “over- or under subscription of accelerators Acc depending on the computation task”). This occurs because “[t]he amount of computing resources required to process a given task can be difficult to predict,” as “[i]t is inevitably difficult to accurately estimate the resource requirements of a job or its constituent tasks because: (i) resource usage of a task varies over time, and (ii) resource usage can vary across tasks of the same job based on the input they process.” EX1006, [0005]. Kambatla addresses this problem by monitoring “actual utilization of resources at a computing node” via a YARN-based resource manager that communicates with node managers at nodes of a cluster, and “improv[ing] effective resource utilization in a distributed computing cluster.” EX1006, [0017], [0042]; *id.*, [0018]-[0023], [0030]-[0036] (describing a central resource manager daemon), [0037]-[0041] (describing a node daemon), [0042]-[0049] (describing “Optimizing

Resource Utilization”). This utilization-based resource allocation for applications deployed to a computer cluster is complementary to the dynamic assignments of Lippert-Budenske. *E.g.*, EX1004, [0015] (teaching “booster allocation is performed as a function of application needs, which means in dependency of processing a specific computation task”); EX1005, 390-391 (describing “input-data dependent” execution of sub-tasks and the remapping of sub-tasks to processors as dynamic parameters evolve over time). Thus, a POSITA seeking to fully leverage the processing capacity of a computer cluster, for example by processing multiple iterative applications concurrently, would naturally seek out and incorporate teachings on cluster resource management daemons that facilitate dynamic resource allocation, such as those taught by Kambatla.

119. *Finally*, a POSITA had a reasonable expectation of success implementing Lippert-Budenske-Kambatla, at least because a POSITA was familiar with and understood how to implement “known resource management techniques (such as in YARN)” (EX1006, [0023]), and because doing so would involve writing only minor software code that was well within the skill of a POSITA. *See, e.g.*, EX1012, xviii (explaining that “YARN makes the effort of building [distributed computing] systems significantly simpler”). Further, a POSITA would have reasonably expected to succeed in incorporating Kambatla’s teachings into Lippert-Budenske because Kambatla explains that its teachings can be successfully

implemented in heterogeneous computing systems comprising “CPU and memory resources” (i.e., computation nodes) and “GPU[s] (graphical processing [units])” (i.e., boosters). EX1006, [0020]; *id.*, [0023], [0059]. Further, the ’442 patent admits that using daemons at nodes to facilitate communication was already known, confirming that a POSITA knew how to implement such features. EX1001, 3:6-18, 3:39-43, 4:32-37.

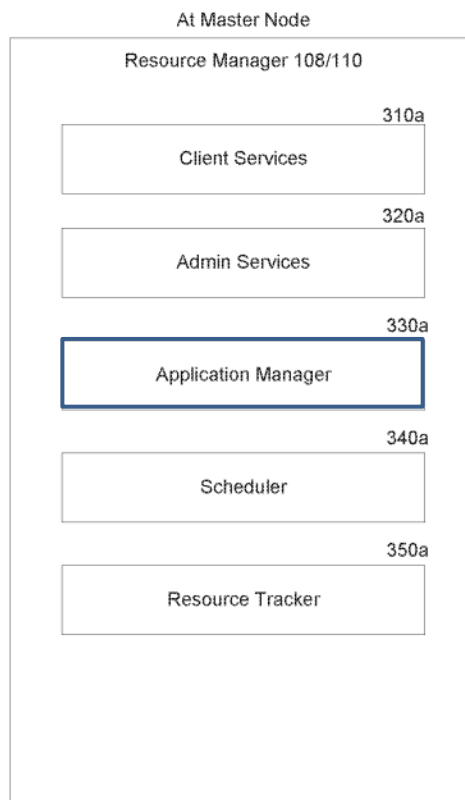
120. Thus, a POSITA implementing the dynamic assignment of sub-tasks between iterations, as taught by Budenske, to computation nodes and boosters of a heterogeneous computing system, as taught by Lippert, would have been motivated to implement a cluster scheduling framework as taught by Kambatla, in order “to improve effective resource utilization” and thus minimize “processing duration.” EX1006, [0042].

1. Claim 2

2. The method according to claim 1, wherein an application manager receives the information and determines the further distribution.
--

121. Lippert-Budenske-Kambatla discloses and teaches this element, and renders this claim obvious for the foregoing and the following reasons. Lippert-Budenske renders claim 1 obvious, as discussed above in Section X.C.1. Further, Lippert-Budenske discloses and teaches this element. Section X.C.2.

122. Moreover, Kambatla expressly discloses various modules of a central resource manager that a POSITA would have understood as the “application manager” recited in the claim. Kambatla discloses an “application manager 330a ... responsible for maintaining a collection of submitted ‘applications’ or ‘jobs.’” EX1006, [0034]. The application manager is a module of a “resource manager daemon.” EX1006, [0031]. The application manager includes an “application master service responsible for communicating with all the application masters at the” computing nodes. EX1006, [0034]. FIG. 3A of Kambatla depicts the resource manager 108 including the application manager 330a.



EX1006, FIG. 3A (annotated). Further, the resource tracker 350a of the resource manager “monitors available resources at the nodes, by receiving status updates from the worker nodes.” EX1006, [0036]; *id.*, [0058] (describing how the resource manager “can actively monitor resource usage of each container at the worker nodes and of the worker nodes themselves,” via “heartbeat information” that includes “utilization information”). At the same time, “[t]he scheduler 340a is responsible for allocating resources to the various applications.” EX1006, [0035]. Despite being depicted as separate components, Kambatla teaches that the resource manager can “include more or fewer components, organized in various configurations.” EX1006, [0030]. A POSITA thus would have understood that the functions of the application manager 330a and the resource tracker 350a were combinable into a single component (*application manager*) that manages an application deployed to cluster resources and communicates with the resources, as well as provides information from the resources to the scheduler 340a of the resource manager to allocate sub-tasks of the application to the resources for processing.

123. In combination, a POSITA implementing Lippert-Budenske-Kambatla would have found it obvious to use an application manager as taught by Kambatla to perform the functions of the Basic Kernel of Budenske, at least because the application manager and the Basic Kernel similarly manage the execution of an application on a plurality of computing devices. Further, a POSITA would have been

motivated to implement the dynamic assignment of sub-tasks to the compute nodes and booster nodes of a heterogeneous computing system, as taught by Lippert-Budenske, with the application manager of Kambatla to efficiently enable and manage the execution of multiple such applications on the plurality of computing resources. *E.g.*, EX1006, [0004], [0023]-[0024], [0034].

124. The POSITA would have reasonably expected to succeed because implementing such an application manager would entail only providing simple software code and configurations to perform the known functions, which was within the skill of a POSITA. *E.g.*, EX1006, [0101] (explaining broadly how “the routines executed to implement the embodiments of the disclosure[] may be implemented”); EX1012, 59-84 (detailing how to install and configure YARN on a cluster).

2. Claim 3

a) Element 3.1

3. The method according to claim 2, wherein a resource manager determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes for the first computing iteration as a function of the computation task and

125. Lippert-Budenske-Kambatla discloses and teaches this element, and renders this claim obvious for the foregoing and following reasons. Section XI.B.1. Lippert-Budenske discloses and teaches this element, as discussed above in Section X.C.3. Further, Kambatla discloses a “central resource manager [that] is a general

resource manager configured to manage and arbitrate resources among applications in the system.” EX1006, [0023], FIGS. 1, 3A. A POSITA implementing Lippert-Budenske-Kambatla would have combined the resource manager of Lippert-Budenske with the central resource manager of Kambatla into a combined resource manager that determines the assignment of the plurality of sub-tasks to the plurality of computation nodes and the plurality of booster nodes for the first computing iteration as a function of the computation task, as taught by Lippert-Budenske. Section X.C.3.a). The POSITA would have done so to gain the additional advantages of Kambatla, namely to enable efficient scheduling of multiple tasks, rather than one computational task, across the processing resources of the heterogeneous computing system, while minimizing “processing duration.” EX1006, [0042], [0004], [0017], [0023]-[0024].

b) Element 3.2

wherein the application manager receives the information and processes the information as input to the resource manager such that the resource manager dynamically alters further distribution during the computing of the computation task.

126. Lippert-Budenske-Kambatla discloses and teaches this element, and renders this claim obvious for the foregoing and the following reasons. Lippert-Budenske-Kambatla includes an *application manager* that receives information from the nodes and processes the information to update an assignment of sub-tasks

to the nodes. Section XI.B.1. Lippert-Budenske-Kambatla further includes a *resource manager*. Section XI.B.2.a). Lippert-Budenske includes a resource manager that dynamically alters further distribution of computation nodes and booster nodes during the computing of the computation task. Section X.C.3.b). Further, Kambatla discloses that its central resource manager includes various modules configured to handle different tasks and feed information to other modules “to aid in resource allocation.” EX1006, [0036], [0031]-[0035]. Kambatla also discloses that “more or fewer components, organized in various configurations ... remain[] within the scope” of Kambatla. EX1006, [0030]. A POSITA thus would have naturally implemented Lippert-Budenske-Kambatla with a central resource manager as taught by Kambatla that includes a module configured to perform the dynamic-assignment functions of Lippert-Budenske’s resource manager, thus achieving the benefits of Kambatla’s resource manager and the dynamic assignment of Lippert-Budenske’s resource manager. Sections X.C.3.b), XI.B. So implemented, the resource manager of Lippert-Budenske-Kambatla uses the information input by the application manager to dynamically alter further distribution during the computing of the computation task, thus combining the functions of the resource manager of Lippert, the HC Kernel Monitor and Basic Kernel of Budenske, and the resource manager of Kambatla.

3. Claim 4

4. The method according to claim 3, wherein the resource manager receives the information such that the resource manager dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other during the computation of the computation task.

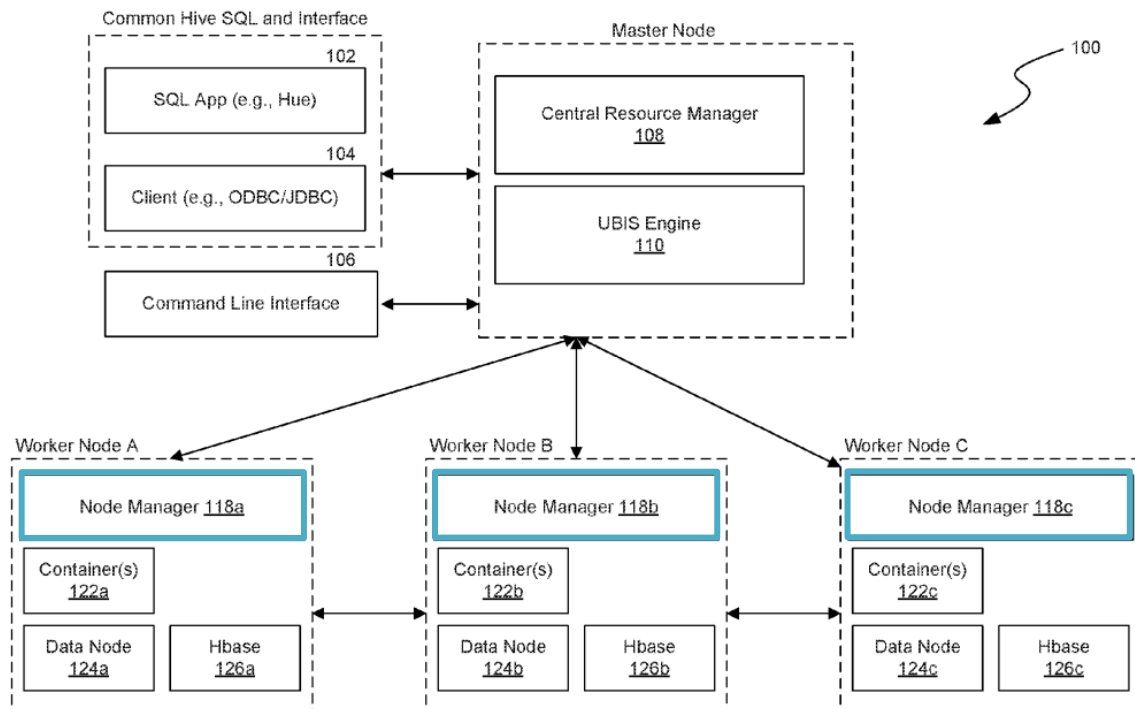
127. Lippert-Budenske-Kambatla teaches and discloses this element, and renders this claim obvious, at least because Lippert-Budenske teaches and discloses a resource manager that receives the information and dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other during the computation of the computation task. Sections XI.B.2, X.C.4. As Lippert-Budenske-Kambatla includes a resource manager that combines the central resource manager of Kambatla with the resource manager of Lippert-Budenske, the resource manager of Lippert-Budenske-Kambatla receives the information and dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other during the computation of the computation task.

4. Claim 5

5. The method according to claim 1, wherein daemons operate in the plurality of computation nodes and the plurality of booster nodes to generate the information.

128. Lippert-Budenske-Kambatla discloses and teaches this element, and renders this claim obvious. Lippert-Budenske renders claim 1 obvious. Section X.C.1. Further, Lippert-Budenske discloses and teaches this element. Section X.C.5.

Kambatla provides further implementation details by expressly disclosing a “resource manager daemon installed at worker nodes in a distributed computing cluster.” EX1006, [0037]. The “resource manager daemon installed at the worker nodes includes a node manager 310b, and one or more application masters 320b.” EX1006, [0038], FIGS. 1, 3B. The central resource manager “[c]ommunicat[es] with node managers 118a-c which act as the agents at each node,” to “allocate and schedule resources available at the various nodes based on the available resources reported from each node manager.” EX1006, [0023].



EX1006, FIG. 1 (annotated). Each node manager “registers with the [central] resource manager and broadcasts the [node’s status] including the status of available resources ... at the node.” EX1006, [0039]. Further, “[a]pplication masters 320b are

responsible for requesting resources from the resource manager 108/110, working with node managers 310b to manage and monitor the containers 330b allocated by the resource manager 108/110.” EX1006, [0041]. Containers are an abstraction of the worker’s nodes computing resources allocated to process a task. EX1006, [0004], [0023].

129. In combination, Lippert-Budenske-Kambatla includes daemons as taught by Kambatla at each worker node—i.e., at each computation node and each booster node—to generate dynamic parameter information as taught by Budenske, as well as information regarding computing capacities and computation task requirements as taught by Lippert. A POSITA would have been motivated to implement Lippert-Budenske-Kambatla in this way because doing so would facilitate efficient communication of the relevant information in a status update for dynamically remapping the sub-tasks among the nodes. In contrast, the “additional information [taught by Budenske and Lippert] will typically add little overhead to an existing node [status update].” EX1006, [0039].

5. Claim 8

8. The method according to claim 3, wherein a daemon operating at a node generates a measure of a loading of the node during processing of a sub-task of the plurality of sub-tasks.
--

130. Lippert-Budenske-Kambatla discloses and teaches this element, and renders this claim obvious. Section XI.B.2. Lippert-Budenske teaches and discloses

this element. Section X.C.8. Further, according to Kambatla, the daemon operating at a node “monitors the containers 330b at the node ... for resource utilization while processes are running” and “calculate[s] the aggregate resource utilization across all containers in the worker node” (*generates a measure of a loading of the node during processing*). EX1006, [0038]-[0040].

131. In combination, Lippert-Budenske-Kambatla includes a daemon operating at each computation node and booster node that generates a measure of a loading of the node during processing of a sub-task of the plurality of sub-tasks. A POSITA would have been motivated to implement Lippert-Budenske-Kambatla in this way, because doing so would allow the dynamic remapping of sub-tasks to the heterogeneous computing system as taught by Budenske to also account for “load balancing” as taught by Lippert. EX1004, [0017].

6. Claim 9

[9.1] A heterogeneous computing system comprising:

132. Lippert-Budenske-Kambatla discloses and teaches this element, at least because Lippert and Budenske each disclose a heterogeneous computing system. Section X.C.1.a).

[9.2] a plurality of computation nodes and a plurality of booster nodes for computing one or more tasks comprising multiple sub-tasks;

133. Lippert-Budenske-Kambatla discloses and teaches this element, at least because Lippert discloses a plurality of computation nodes and a plurality of booster nodes for computing one or more tasks comprising multiple sub-tasks. Section X.C.1.a).

[9.3] a communication interface connecting the plurality of computation nodes with each other and the plurality of booster nodes;

134. Lippert-Budenske-Kambatla discloses and teaches this element, at least because Lippert-Budenske discloses and teaches this element. Section X.C.9. Kambatla also discloses a “cluster of worker nodes in [] communication (e.g., via a computer network) with each other and one or more master nodes.” EX1006, [0021].

135. In combination, Lippert-Budenske-Kambatla includes a communication interface connecting the plurality of computation nodes with each other and the plurality of booster nodes.

[9.4] a resource manager for assigning at least a portion of the plurality of booster nodes and at least a portion of the plurality of computation nodes to each other for the computing of the one or more tasks in a first computing iteration; and

136. Lippert-Budenske-Kambatla discloses and teaches this element. Sections X.C.1, XI.B.1, XI.B.2.

[9.5] an application manager configured to receive information from daemons operating in at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes to update a distribution of the multiple sub-tasks between the plurality of computation nodes and the plurality of booster nodes in a further computing iteration.

137. Lippert-Budenske-Kambatla discloses and teaches this element.

Sections X.C.1, XI.B.1, XI.B.4.

7. **Claim 10**

10. The heterogeneous computing system according to claim 9, wherein the resource manager receives the information such that the resource manager dynamically alters assignment of the plurality of computation nodes and the plurality of booster nodes to each other.

138. Lippert-Budenske-Kambatla discloses and teaches this claim. Sections XI.B.6, XI.B.3.

I, Darrell Long, do hereby declare and state, that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: December 20, 2024



Darrell D. E. Long, Ph.D.

APPENDIX A

Cumulative Biobibliography
University of California, Santa Cruz

April 27, 2024

DARRELL DON EARL LONG

Distinguished Research Professor of Engineering
Kumar Malavalli Endowed Chair Professor in Storage Systems Research, Emeritus

Employment History

2024–	Distinguished Visiting Scholar, Santa Clara University
2023–	Distinguished Research Professor of Engineering, University of California, Santa Cruz Distinguished Professor of Engineering, <i>Emeritus</i> , University of California, Santa Cruz
2019–23	Director, Center for Research Systems & Storage, University of California, Santa Cruz
2017–23	Distinguished Professor of Engineering, University of California, Santa Cruz
2005–23	Kumar Malavalli Endowed Chair Professor in Storage Systems Research
2004–10	Associate Dean for Research and Graduate Studies, Jack Baskin School of Engineering, University of California, Santa Cruz
2001–19	Director, Storage Systems Research Center, University of California, Santa Cruz
1999–17	Professor, Computer Science, University of California, Santa Cruz
1998–01	Associate Dean, Jack Baskin School of Engineering, University of California, Santa Cruz
1994–99	Associate Professor, Computer Science, University of California, Santa Cruz
1988–94	Assistant Professor, Computer Science, University of California, Santa Cruz
1986–88	Research Assistant, Computer Science & Engineering, University of California, San Diego
1985–87	Teaching Associate, Computer Science & Engineering, University of California, San Diego
1984–87	Lecturer in Mathematics, Department of Mathematical Sciences, San Diego State University
1981–84	Systems Programmer, University Computer Center, San Diego State University

Visitor History

2019 (Winter)	Professeur Invité, Sorbonne Université
2016–	Associate Member, European Organization for Nuclear Research (CERN)
2016 (Fall)	Professeur Invité, Conservatoire National des Arts et Métiers
2014–2020	Visiting Scientist, Lawrence Livermore National Laboratory
2014 (Fall)	Professeur Invité, Conservatoire National des Arts et Métiers Professeur Invité, Université Paris–Descartes

2013 (Winter)	Professeur Invité, Conservatoire National des Arts et Métiers Professeur Invité, Université Paris–Descartes Professeur Invité, Université Paris–Dauphine
2012–2014	Visiting Professor, United States Naval Postgraduate School
2011 (Winter)	Professeur Invité, Conservatoire National des Arts et Métiers
2010 (Winter)	Professeur Invité, Université Paris–Dauphine
2009 (Winter)	Professeur Invité, Université Paris–Dauphine
2008 (Winter)	Visiting Professor, University of Technology, Sydney
2007 (Winter)	Visiting Scholar, University of California, San Diego Visiting Scholar, Center for Communications Research
1995–2011	Visiting Scientist, IBM Almaden Research Center

Education

Ph.D.	1988	University of California, San Diego, Computer Science
M.S.	1986	University of California, San Diego, Computer Science
B.S.	1984	San Diego State University, Computer Science

Honors and Awards

2018	IBM Faculty Award
2016	Best Paper Award, “RESAR: Reliable Storage at Exabyte Scale,” <i>Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems</i>
2015	Best Paper Award, “Classifying Data to Reduce Long Term Data Movement in Shingled Write Disks,” <i>Conference on Mass Storage Systems and Technologies</i>
2013	Best Short Paper Award, “A File By Any Other Name: Managing File Names with Metadata,” <i>International Systems and Storage Conference</i>
2012	Certificate of Appreciation for Outstanding Service, National Research Council IEEE Certificate of Appreciation
2011	Chancellor’s Achievement Award for Diversity
2010	Professor <i>ad Honorem</i> de la Universidad Católica del Uruguay
2008	Fellow, American Association for the Advancement of Science Certificate of Appreciation for Outstanding Service, National Research Council
2006	Fellow, Institute of Electrical and Electronics Engineers
2005	Kumar Malavalli Endowed Chair in Storage Systems Research IBM Research Invention Achievement Award (Third Plateau) Certificate of Appreciation for Outstanding Service, National Research Council

2003	IBM Faculty Award
2002	IBM Research Invention Achievement Award (Second Plateau) IEEE Computer Society Certificate of Appreciation
2001	IBM Corporate Accomplishment Award for Adaptive Differential Back-up in the Tivoli Storage Manager
1997	IBM Research Invention Achievement Award (First Plateau)
1996	IBM Research Invention Achievement Award IEEE Computer Society Certificate of Appreciation
1995	Honorable mention, 1994–95 Excellence in Teaching Award Best Paper Award, “A Longitudinal Study of Internet Host Reliability,” <i>Symposium on Reliable Distributed Systems</i>
1994	Senior Member, Institute of Electrical and Electronics Engineers
1993	IEEE Computer Society Certificate of Appreciation
1992	Regents Junior Faculty Fellow
1991	Student Alumni Council Favorite Professor Award
1989	Regents Junior Faculty Fellow
1988	Regents Junior Faculty Fellow

Grants

2022–25	<i>Trust Worthy Information Storage Technology Enhanced Devices (TWISTED)</i> , Defense Advanced Research Projects Agency (DARPA), Phase 2: \$2,500,000 (with P. Alvaro). Panel Reviewed.
2020–21	<i>Trust Worthy Information Storage Technology Enhanced Devices (TWISTED)</i> , Defense Advanced Research Projects Agency (DARPA), Phase 1: \$225,000. Panel Reviewed.
2019–24	<i>Phase II I/UCRC CRSS: Center for Research in Storage Systems</i> , National Science Foundation, \$500,000. Peer Reviewed.
2018–21	<i>A Multi-Layered Deniable Steganographic File System</i> , National Science Foundation, \$500,000. Peer Reviewed.
2017–20	<i>Integrated End-to-end Performance Prediction and Diagnosis for Extreme Scientific Workflows</i> , Department of Energy, Office of Science, \$300,000. Panel Reviewed.
2015–17	<i>Automatic Tuning and Contention Management for Lustre</i> , Intel Corporation, \$187,024. Panel Reviewed.
2015–18	<i>Automatic Storage and Network Contention Management for Large-scale High-Performance Computing Systems</i> , National Science Foundation, \$450,000. Peer Reviewed.
2013–18	<i>I/UCRC: A Single-Site I/UCRC Center for Research in Storage Systems</i> , National Science Foundation, \$468,850 (with E. Miller). Peer Reviewed.
2012–13	<i>Better File System Management Through Rich Metadata and Provenance</i> , NASA, \$40,000. Panel Reviewed.

- 2012–15 *RESAR: Robust, Efficient, Scalable, Autonomous Reliable Storage for the Cloud*, National Science Foundation, \$374,700. **Peer Reviewed.**
- 2011–12 *Reducing Fragmentation in Deduplicated Storage Systems*, Hewlett-Packard Laboratories, \$50,000. **Panel Reviewed.**
Better File System Management Through Rich Metadata and Provenance, NASA, \$40,000. **Panel Reviewed.**
Improving Automated Metadata Extraction, Analysis and Reporting, United States Navy, \$150,000. **Panel Reviewed.**
- 2010–11 *Metadata Exploration in Digital Forensics*, United States Navy, \$196,156. **Panel Reviewed.**
- 2010–13 *Dynamic Non-Hierarchical File Systems for Exascale Storage*, Department of Energy, Office of Science, \$1,462,106. **Peer Reviewed.**
- 2010–13 *LockBox: Enabling Users to Keep Data Safe*, National Science Foundation, \$499,907 (with E. Miller). **Peer Reviewed.**
- 2009–14 *Collaborative Research: A Multi-University IIUCRC Center on Intelligent Storage*, National Science Foundation, \$275,000 (with E. Miller). **Peer Reviewed.**
- 2009–12 *Scalable Data Management Using Metadata and Provenance*, National Science Foundation, \$553,000 (with E. Miller, in collaboration with Margo Seltzer, Harvard University). **Peer Reviewed.**
- 2010–11 *Trading Storage for Computation*, NASA, \$61,000. **Panel Reviewed.**
- 2009–10 *Development of a Collaborative Project for Remotely Sensed Science and Technology*, NASA, \$300,000 (with Raphael Kudela, Ethan Miller, Roberto Manduci, Donald Potts, Eli Silver, Michael Loik and Chris Wilmers). **Panel Reviewed.**
- 2007 *ViewFS: Dynamic Name-Spaces for Metadata-Rich File Systems*, Lawrence Livermore National Laboratory, \$74,994 (with E. Miller). **Panel Reviewed.**
- 2006–09 *End-to-End Performance Management for Large Distributed Storage*, National Science Foundation, \$956,647 (with S. Brandt). **Peer Reviewed.**
- 2006–11 *Petascale Data Storage Institute*, Department of Energy, Office of Science, \$11,250,000 (in collaboration with Carnegie-Mellon University, University of Michigan, and the Department of Energy National Laboratories). **Peer Reviewed.**
- 2005–10 *Institute for Scalable Scientific Data Management*, Department of Energy, Los Alamos National Laboratory, \$4,750,000 (with S. Brandt and E. Miller). **Panel Reviewed.**
- 2005–06 *Scalable File Systems for High Performance Computing*, Department of Energy, Defense Programs Laboratories, \$250,000 (with S. Brandt, E. Miller, M. Abadi and C. Maltzahn). **Peer Reviewed.**
- 2003 *A Scalable On-Line Associative Deep Store*, University of California, Microelectronics Electronics Innovation and Computer Research Opportunities Program (MICRO), \$27,000. **Panel Reviewed.**
- 2003–06 *A Scalable On-Line Associative Deep Store*, National Science Foundation, \$294,012. **Peer Reviewed.**
- 2002–07 *Applications of Data Grouping for Effective Mobility*, National Science Foundation, \$338,130. **Peer Reviewed.**
- 2002–05 *Scalable File Systems for High Performance Computing*, Department of Energy, Defense Programs Laboratories, \$900,000 (with S. Brandt, E. Miller and K. Obraczka). **Panel Reviewed.**

- 2001–02 *Building a High-Performance Storage Systems from Commodity Components*, Department of Energy, Lawrence Livermore National Laboratory, \$65,000. **Panel Reviewed.**
- 2000–01 *Application of Probe-based Storage to High Performance Computing*, Department of Energy, Lawrence Livermore National Laboratory, \$39,239. **Panel Reviewed.**
- 2000–03 *Architectures and Algorithms to Exploit Probe-Based Storage*, National Science Foundation, \$345,191 (with S. Brandt and T. Madhyastha). **Peer Reviewed.**
- 2000–02 *An Experimental Study of Broadcasting Protocols for Video-on-Demand*, National Science Foundation, \$100,000. **Peer Reviewed.**
- 1999–01 *High Performance Integration of Advanced Tertiary Stores*, National Science Foundation, \$170,353. **Peer Reviewed.**
- 1998 *Real-Time Environmental Information Network and Analysis System (REINAS)*, Office of Naval Research, \$100,000. **Panel Reviewed.**
- 1998–99 *Tactical Environmental Data System/Real-time (TEDS/RT)*, Naval Research Laboratory, \$75,000. **Panel Reviewed**
- 1997–02 *National Partnership for Advanced Computing Infrastructure (NPACI)*, National Science Foundation, \$819,000 (with P. Mantey, A. Pang and S. Flatté). **Panel Reviewed.**
- 1997–99 *Predictive Power Conservation*, National Science Foundation, \$122,011 (with D. Helmbold). **Peer Reviewed.**
- 1997–99 *Improving Cache Performance by Predicting I/O System Actions*, National Science Foundation, \$136,647. **Peer Reviewed.**
- 1996–97 *Efficient Back-up and Restore Using Differential Files*, International Business Machines Corporation, \$79,864.
- 1995–98 *Instant Infrastructure and Distributed Resource Management*, Office of Naval Research, \$90,911. **Panel Reviewed.**
- 1995–96 *Efficient Resource Utilization Through Idle-Time Prediction*, University of California Seed Funds, \$15,000 (with D. Helmbold).
- 1993–94 *Global-scale Distributed Data Base Management*, Academic Senate Committee on Research, \$1,500.
- 1993–95 *Integration of Heterogeneous Real-time Data Repositories for Scientific Use*, Office of Naval Research, \$219,164. **Panel Reviewed.**
- 1992–93 *Estimating Host Reliability Using the Internet*, University of California, Microelectronics Electronics Innovation and Computer Research Opportunities Program (MICRO), \$5,481. **Panel Reviewed.**
- 1992 University of California, Regents Junior Faculty Fellowship, \$700.
- 1992–97 *Real-time Environmental Information Network and Analysis System (REINAS)*, Office of Naval Research, \$4,767,112 (with P. Mantey *et al.*). **Peer Reviewed.**
- 1992–93 University of California, Regents Junior Faculty Fellowship, \$750.
- 1992–93 *Estimating Reliability Using the Internet*, University of California Seed Funds, \$10,000.
- 1992–93 *Estimating Host Reliability Using the Internet*, Sun Microsystems, Incorporated, \$16,590. **Panel Reviewed.**

- 1991–92 *A Distributed Architecture for High-speed I/O*, Department of Energy, Lawrence Livermore National Laboratory, \$15,183. **Panel Reviewed.**
- 1991–93 *High-speed Distributed Storage Management*, National Science Foundation, \$68,955. **Peer Reviewed.**
- 1991–92 *A High-speed System to Support Multimedia*, Academic Senate Committee on Research, \$1,500.
- 1991–92 *Instructional Improvement Grant*, Academic Senate Committee on Teaching, \$200.
- 1990–91 *Concurrent Systems Laboratory*, University of California Seed Funds, \$12,500.
- 1989–90 *Fault-tolerant Operating System Techniques*, University of California, Microelectronics Electronics Innovation and Computer Research Opportunities Program, (MICRO) \$4,000. **Panel Reviewed.**
- 1989 University of California, Regents Junior Faculty Fellowship, \$500.
- 1988–89 *Modeling Fault-tolerant Computations*, Academic Senate Committee on Research, \$1,500.
- 1988 University of California, Regents Junior Faculty Fellowship, \$1,000.

Scholarly and Creative Work

Books

1. Ruth A. David, Steven R.J. Brueck, Stephen W. Drew, Alan H. Epstein, Robert A. Furman, Sharon Glotzer, Christopher C. Green, Diane E. Griffin, J. Jerome Holton, Michael R. Ladisch, Darrell Long, Frederick R. Lopez, Richard M. Osgood, Stewart D. Personick, Alton D. Romig, S. Shankar Sasrty, James B. Smith, Camillo J. Taylor and Diane Wiley. *Avoiding Surprise in an Era of Global Technology Advances*. National Research Council, National Academies Press, 2005. **Peer Reviewed.**
2. John L. Carroll and Darrell D. E. Long. *Theory of Finite Automata*. Englewood Cliffs, New Jersey: Prentice-Hall, 1989. **Editorial Review.**

Chapters in Books

1. Ismail Ari, Ahmed Amer, Ethan L. Miller, Robert Gramacy, Scott A. Brandt and Darrell D. E. Long. “ACME: Adaptive Caching using Multiple Experts,” *Distributed Data & Structures 4* (Proceedings in Informatics 14), W. Litwin and G. Lèvy, Eds. Paris, France: Carleton Scientific, March 2002, pp. 143–158. **Peer Reviewed.**
2. Steven W. Carter, Darrell D. E. Long and Jehan-François Pâris. “Video-on-Demand Broadcasting Protocols,” *Multimedia Communications: Directions and Innovations*, J. D. Gibson, Ed. San Diego: Academic Press, 2000, pp. 179–189. **Peer Reviewed.**
3. Richard A. Golding and Darrell D. E. Long. “Simulation Modeling of Weak-Consistency Protocols,” *Network Systems Design*, E. Gelenbe, K. Bagchi and G. Zobrist, Eds. Gordon and Breach Science Publishers, 1999, pp. 161–186. **Invited, Peer Reviewed.**
4. Darrell D. E. Long and John L. Carroll. “Using Simulation to Evaluate Consistency Protocols for Replicated Data,” *Progress in Simulation II*, G. W. Zobrist and J. V. Leonard, Eds. Ablex Publishing, 1995. **Invited, Peer Reviewed.**

Peer Reviewed Journals

1. Liangkang Zhang, Yulai Xie, Minpeng Jin, Pan Zhou, Gongming Xu, Yafeng Wu, Dan Feng, Dan and Darrell Long. "A Novel Hybrid Model for Docker Container Workload Prediction," *IEEE Transactions on Network and Service Management*, to appear.
2. Yafeng Wu, Yulai Xie, Xuelong Liao, Pan Zhou, Dan Feng, Lin Wu, Xuan Li, Wildani, Avani and Darrell Long. "Paradise: Real-time, Generalized, and Distributed Provenance-Based Intrusion Detection," *ACM Transactions on Dependable and Secure Computing*, to appear.
3. Daniel Bittman, Peter Alvaro, Pankaj Mehra, Darrell D. E. Long and Ethan L. Miller. "Twizzler: A Data-centric OS for Non-volatile Memory," *ACM Transactions on Storage*, vol. 17, no. 2, June 2021.
4. Austen Barker, Yash Gupta, James Hughes, Ethan L. Miller and Darrell D. E. Long. "Rethinking the Adversary and Operational Characteristics of Deniable Storage," *Journal of Surveillance, Security and Safety*, vol. 2, May 27, 2021.
5. Yulai Xie, Minpeng Jin, Zhuping Zou, Gongming Xu, Dan Feng, Wenmao Liu and Darrell Long. "Real-time Prediction of Docker Container Resource Load Based on A Hybrid Model of ARIMA and Triple Exponential Smoothing," *IEEE Transactions on Cloud Computing*, April 22, 2020.
6. Yulai Xie, Yafeng Wu, Dan Feng and Darrell Long. "P-Gaussian: Provenance-Based Gaussian Distribution for Detecting Intrusion Behavior Variants Using High Efficient and Real Time Memory Databases," *IEEE Transactions on Dependable and Secure Computing*, December 17, 2019.
7. Yulai Xie, Zhuping Zhou, Kai Huang, Gongming Xu, Dan Feng and Darrell Long. "A Docker Container Anomaly Monitoring System Based on Optimized Isolation Forest," *IEEE Transactions on Cloud Computing*, September, 2018.
8. Die Hu, Dan Feng, Yulai Xie, Gongming Xu, Xinrui Gu and Darrell Long. "Efficient Provenance Management via Clustering and Hybrid Storage in Big Data Environments," *IEEE Transactions on Big Data*, December 2020, vo. 6, no. 4, pp. 792–803.
9. Yulai Xie, Dan Feng, Yuchong Hu, Yan Li, Staunton Sample and Darrell Long. "Pagoda: A Hybrid Approach to Enable Efficient Real-time Provenance Based Intrusion Detection in Big Data Environments," *IEEE Transactions on Dependable and Secure Computing*, August 29, 2018.
10. Stephanie N. Jones, Ahmed Amer, Ethan L. Miller, Darrell D. E. Long, Rekha Pitchumani and Christina R. Strong. "Classifying Data to Reduce Long-Term Data Movement in Shingled Write Disks," *ACM Transactions on Storage*, vol. 12 no. 1, February 2016.
11. Yulai Xie, Dan Feng, Yan Li and Darrell D. E. Long. "Oasis: an Active Storage Framework for Object Storage Platform," *Future Generation Computer Systems*, vol. 56, March 2016, pp. 746–758.
12. Nicolás López, Matías Rodríguez, Catalina Fellegi, Darrell Long and Thomas Schwarz, S.J. "Even or Odd: A Simple Graphical Authentication System (*Par o Impar: un sistema gráfico simple de autenticación*)," *IEEE Latin America Transactions* (Revista IEEE América Latina), 13(3), pp. 804–809, March 2015.
13. Yulai Xie, Kiran-Kumar Muniswamy-Reddy, Dan Feng, Yan Li and Darrell D. E. Long. "Evaluation of a Hybrid Approach for Efficient Provenance Storage," *ACM Transactions on Storage*, vol. 9 no. 4, November 2013.
14. Witold Litwin, Darrell Long and Thomas Schwarz. "Combining Chunk Boundary and Chunk Signature Calculations for Deduplication (*Combinar la calculacion de limites de trozos y de firmas de trozos para deduplicacion*)," *IEEE Latin America Transactions* (Revista IEEE América Latina), 10(1), pp. 1305–1311, January 2012.

15. Ahmed Amer, JoAnne Holliday, Darrell D. E. Long, Ethan L. Miller, Jehan-François Pâris and Thomas Schwarz, S.J. "Data Management and Layout for Shingled Magnetic Recording," *IEEE Transactions on Magnetics*, vol. 47, no. 10, pp. 3691–3697, October 2011.
16. Lawrence L. You, Kristal T. Pollack, Darrell D. E. Long and K. Gopinath. "PRESIDIO: A Framework for Efficient Archival Data Storage," *ACM Transactions on Storage*, vol. 7, no. 2, art. 6, pp. 1–60, July 2011.
17. Bo Hong, Scott A. Brandt, Darrell D. E. Long, Ethan L. Miller and Ying Lin. "Using MEMS-Based Storage in Computer Systems – Device Modeling and Management," *ACM Transactions on Storage*, vol. 2, no. 2, May 2006, pp. 139–160.
18. Bo Hong, Feng Wang, Scott A. Brandt, Darrell D. E. Long and Thomas J. E. Schwarz, S.J. "Using MEMS-Based Storage in Computer Systems – MEMS Storage Architectures," *ACM Transactions on Storage*, vol. 2, no. 1, February 2006, pp. 1–21.
19. Randal Burns, Larry Stockmeyer and Darrell Long. "In-Place Reconstruction of Version Differences," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 4, July 2003, pp. 973–984.
20. Miklós Ajtai, Randal Burns, Ronald Fagin, Darrell Long and Larry Stockmeyer. "Compactly Encoding Unstructured Inputs with Differential Compression," *Journal of the ACM*, vol. 49, no. 3, May 2002, pp. 318–367.
21. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "Efficient Data Distribution in a Web Server Farm," *IEEE Internet Computing*, vol. 5, no. 4, July 2001, pp. 56–65.
22. Randal C. Burns, Robert M. Rees, Larry Stockmeyer, and Darrell D. E. Long. "Scalable Session Locking for a Distributed File System," *Cluster Computing Journal*, Kluwer Academic Publishers, vol. 4, no. 4, October 2001, pp. 295–306.
23. David P. Helmbold, Darrell D. E. Long, Tracey L. Sconyers and Bruce Sherrod. "Adaptive Disk Spin-Down for Mobile Computers," *ACM/Baltzer Mobile Networks and Applications Journal*, vol. 5, no. 4 (2000), pp. 285–297.
24. Benjamin C. Reed, Darrell D. E. Long, Edward G. Chron and Randal C. Burns. "Authenticating Network Attached Storage," *IEEE Micro*, vol. 20, no. 1, January 2000, pp. 49–57.
25. Steven W. Carter and Darrell D. E. Long. "Improving Bandwidth Efficiency of Video-on-Demand Servers," *Computer Networks*, Elsevier, vol. 31, nos. 1–2, January 1999, pp. 99–111.
26. Eric C. Rosen, Theodore R. Haining, Darrell D. E. Long, Patrick E. Mantey. "REINAS: A Real-time System for Managing Environmental Data," *Journal of Software Engineering and Knowledge Engineering*, vol. 8, no. 1 (1998), pp. 35–53.
27. Darrell D. E. Long, Bruce R. Montague and Luis-Felipe Cabrera. "Swift/RAID: A Distributed RAID System," *Computing Systems*, vol. 7, no. 3 (1994), pp. 333–359.
28. Ivan Fellner, Ivan Racko, Milos Racek, Karol Fabian and Darrell D. E. Long. "A Comparison of Two Implementations of the Token Ring Priority Function," *International Journal in Computer Simulation*, vol. 5, no. 1 (1994), pp. 13–28.
29. Luis-Felipe Cabrera and Darrell D. E. Long. "Swift: Using Distributed Disk Striping to Provide High I/O Data Rates," *Computing Systems*, vol. 4, no. 4 (1991), pp. 405–436.
30. Richard A. Golding and Darrell D. E. Long. "Accessing Replicated Data in a Large-Scale Distributed System," *International Journal in Computer Simulation*, vol. 1, no. 4 (1991), pp. 347–372.
31. Jehan-François Pâris and Darrell D. E. Long. "The Performance of Available Copy Protocols for the Management of Replicated Data," *Performance Evaluation*, vol. 11, no. 1, April 1990, pp. 9–30.
32. Darrell D. E. Long, John L. Carroll and Kris Stewart. "Estimating the Reliability of Regeneration-Based Replica Control Protocols," *IEEE Transactions on Computers*, vol. 38, no. 12, (December 1989), pp. 1691–1702.

Peer Reviewed Conference Papers

1. Eugene Chou, Leo Conrad-Shah, Austen Barker, Andrew Quinn, Ethan L. Miller and Darrell D. E. Long. "Lethe: Secure Deletion by Addition," *Proceedings of the Workshop on Challenges and Opportunities of Efficient and Performant Storage Systems* (CHEOPS), Rome, ACM: May 8, 2023.
2. Kyle Fredrickson, Austen Barker and Darrell Long. "A Multiple Snapshot Attack on Deniable Storage Systems," *Proceedings of the Twenty-ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2021), IEEE, November 3–5, 2021.
3. Oceane Bel, Kenneth Chang, Nathan R. Tallent, Dirk Duellmann, Ethan L. Miller, Faisal Nawab and Darrell D. E. Long. "Geomancy: Automated Performance Enhancement through Data Layout Optimization," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, October 26–30, 2020.
4. James Byron, Ethan L. Miller and Darrell D. E. Long. "Measuring the Cost of Reliability in Archival Systems," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, October 26–30, 2020.
5. Austen Barker, Yash Gupta, Sabrina Au, Eugene Chou, Ethan L. Miller and Darrell D. E. Long. "Artifice: Data in Disguise," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, October 26–30, 2020.
6. Daniel Bittman, Peter Alvaro, Darrell Long and Ethan Miller. "Twizzler: a Data-Centric OS for Non-Volatile Memory," *Proceedings of the 2020 USENIX Annual Technical Conference*, Boston: Usenix Association, July 15–17, 2020.
7. Daniel Bittman, Peter Alvaro, Darrell D. E. Long and Ethan L. Miller. "A Tale of Two Abstractions: The Case for Object Space," *Proceedings of the Eleventh Hot Topics in Storage and File Systems* (Hot Storage 2019), Renton: Usenix Association, July 8–9, 2019.
8. Daniel Bittman, Darrell D. E. Long, Peter Alvaro, and Ethan L. Miller. "Optimizing Systems for Byte-Addressable NVM by Reducing Bit Flipping," *Proceedings of the Conference on File and Storage Technologies* (FAST), San Jose: Usenix Association, February 25–28, 2019.
9. Sinjoni Mukhopadhyay, Joel Frank, Daniel Bittman, Darrell Long and Ethan Miller. "Efficient Reconstruction Techniques for Disaster Recovery in Secret-Split Datastores," *Proceedings of the Twenty-sixth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2018), Milwaukee: IEEE, September 25–28, 2018.
10. James Byron, Ethan Miller and Darrell Long. "Using Simulation to Design Scalable and Cost-Efficient Archival Storage Systems," *Proceedings of the Twenty-sixth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2018), Milwaukee: IEEE, September 25–28, 2018.
11. Daniel Bittman, Matthew Gray, Justin Raizes, Sinjoni Mukhopadhyay, Matthew Bryson, Peter Alvaro, Darrell Long and Ethan Miller. "Designing Data Structures to Minimize Bit Flips on NVM," *Proceedings of the Seventh IEEE Non-Volatile Memory Systems and Applications Symposium* (NVMSA '18) Hakodate, Japan: IEEE, August 28–31, 2018.
12. Oceane Bel, Kenneth Chang, Daniel Bittman, Ethan L. Miller, Darrell D. E. Long and Hiroshi Isozaki. "Inkpack: A Secure, Data-Exposure Resistant Storage System," *Proceedings of the Eleventh Annual International Systems and Storage Conference* (SYSTOR), Haifa, Israel: ACM, June 4–8, 2018.

13. Yan Li, Kenneth Chang, Oceane Bel, Ethan L. Miller and Darrell D. E. Long. "CAPES: Unsupervised Storage Performance Tuning Using Neural Network-Based Deep Reinforcement Learning," *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (SC '17), November 13–16, 2017.
14. Carlos A. Rincon, Jehan-François Pâris, Ricardo Vilalta, Albert M. K. Cheng and Darrell D. E. Long. "Disk Failure Prediction in Heterogeneous Environments," *Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, Seattle: IEEE, July 9–12, 2017.
15. Thomas Schwarz, Ahmed Amer, Thomas Kroeger, Ethan Miller, Darrell Long and Jehan-François Pâris. "RE-SAR: Reliable Storage at Exabyte Scale," *Proceedings of the Twenty-fourth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2016), London: IEEE, September 21–23, 2016.
16. Yan Li, Ethan Miller, Darrell Long and Yash Gupta. "Pilot: A Framework that Understands How to Do Performance Benchmarks the Right Way," *Proceedings of the Twenty-fourth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2016), London: IEEE, September 21–23, 2016.
17. Jehan-François Pâris and Darrell D. E. Long. "Pirogue: A lighter dynamic version of the Raft distributed consensus algorithm," *Proceedings of the Thirty-fourth International Performance of Computers and Communication Conference* (IPCCC 2015), Nanjing, China: IEEE, December 14–16, 2015.
18. Thomas Schwarz, Darrell D. E. Long and Jehan-François Pâris. "Triple failure tolerant storage systems using only exclusive-or parity calculations," *Proceedings of the Twenty-first IEEE Pacific Rim International Symposium on Dependable Computing* (PRDC 2015), Zhangjiajie, China: IEEE, November 18–20, 2015.
19. Jehan-François Pâris and Darrell Long. "Reducing the Energy Footprint of a Distributed Consensus Algorithm," *Proceedings of the European Dependable Computing Conference* (EDCC 2015), Paris: IEEE, September 7–11, 2015.
20. Stephanie N. Jones, Ahmed Amer, Ethan L. Miller, Darrell D. E. Long, Rekha Pitchumani and Christina R. Strong. "Classifying Data to Reduce Long Term Data Movement in Shingled Write Disks," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, May 30–June 5, 2015.
21. Joel C. Frank, Shayna M. Frank, Lincoln A. Thurlow, Thomas M. Kroeger, Ethan L. Miller and Darrell D. E. Long. "Percival: A Searchable Secret Split Datastore," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, May 30–June 5, 2015.
22. Yan Li, Xiaoyuan Lu, Ethan L. Miller and Darrell D. E. Long. "ASCAR: Automating Contention Management for High-Performance Storage Systems," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Santa Clara University: IEEE, May 30–June 5, 2015.
23. Jehan-François Pâris, Ahmed Amer, Darrell Long and Thomas Schwarz. "Self-Repairing Disk Arrays," *Proceedings of the Fifth International Workshop on Adaptive Self-tuning Computing Systems* (ADAPT), January 21, 2015, Amsterdam, The Netherlands.
24. Avani Wildani, Ethan L. Miller, Ian F. Adams and Darrell D. E. Long. "PERSES: Data Layout for Low Impact Failures," *Proceedings of the Twenty-second International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2014), Paris: IEEE, September 9–11, 2014.
25. Yan Li and Darrell D. E. Long. "Which Storage Device is Greenest? Modeling the Energy Cost of I/O Workloads," *Proceedings of the Twenty-second International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2014), Paris: IEEE, September 9–11, 2014.

26. Jehan-François Pâris, Thomas J. E. Schwarz, Ahmed Amer and Darrell Long. "Protecting RAID Arrays Against Unexpectedly High Disk Failure Rates," *Proceedings of the Twentieth IEEE Pacific Rim International Symposium on Dependable Computing* (PRDC 2014), Singapore: IEEE, November 18–21, 2014.
27. Alex Nelson, Erik Steggall and Darrell Long. "Cooperative mode: Comparing interpretations of the Xbox 360 storage system," *Proceedings of the Digital Forensics Research Conference* (DFRWS), Denver: ACM, August 3–6, 2014.
28. Soror Sahri, Rim Moussa, Salima Benbernou and Darrell Long. "DBaaS-Expert: A Recommender for the Selection of the Right Cloud Database," *Proceedings of the Twenty-first International Symposium on Methodologies for Intelligent Systems* (ISMIS), Roskilde, Denmark: Springer, June 25–27, 2014.
29. Aleatha Parker-Wood, Darrell D. E. Long, Ethan Miller, Philippe Rigaux and Andy Isaacson. "A File By Any Other Name: Managing File Names with Metadata," *Proceedings of the Seventh Annual International Systems and Storage Conference* (SYSTOR), Haifa, Israel: ACM, June 10–12, 2014.
30. Ahmed Amer, Darrell D. E. Long and Thomas Schwarz. "Reliability Challenges for Storing Exabytes," *Proceedings of the International Conference on Computing, Networking and Communications* (ICNC), Honolulu: IEEE, February 3–6, 2014, pp. 907–913.
31. Jehan-François Pâris, Darrell D. E. Long and Thomas Schwarz, S.J. "Zero-Maintenance Disk Arrays," *Proceedings of the IEEE Nineteenth Pacific Rim International Symposium on Dependable Computing* (PRDC), Vancouver, British Columbia, Canada: IEEE, December 2–4, 2013.
32. Thomas Schwarz, Darrell D. E. Long and Jehan-François Pâris. "Reliability of Disk Arrays with Double Parity," *Proceedings of the IEEE Nineteenth Pacific Rim International Symposium on Dependable Computing* (PRDC), Vancouver, British Columbia, Canada: IEEE, December 2–4, 2013.
33. Jehan-François Pâris, Darrell D. E. Long and Witold Litwin. "Three-Dimensional Redundancy Codes for Archival Storage," *Proceedings of the Twenty-first International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS 2013), San Francisco: IEEE, August 14–16, 2013.
34. Aleatha Parker-Wood, Brian Madden, Michael McThrow, Darrell D.E. Long, Ian F. Adams and Avani Wildani. "Examining Extended and Scientific Metadata for Scalable Index Designs," *Proceedings of the Sixth Annual International Systems and Storage Conference* (SYSTOR), Haifa, Israel: ACM, June 10–12, 2013.
35. Hsu-Wan Kao, Jehan-François Pâris, Thomas Schwarz, S.J. and Darrell D. E. Long. "A Flexible Simulation Tool for Estimating Data Loss Risks in Storage Arrays," *Proceedings of the Conference on Mass Storage Systems and Technologies* (MSST), Long Beach: IEEE, May 9, 2013.
36. Yan Li, Nakul Sanjay Dhotre, Yasuhiro Ohara, Thomas M. Kroeger, Ethan L. Miller and Darrell D. E. Long. "Horus: Fine-Grained Encryption-Based Security for Large-Scale Storage," *Proceedings of the Conference on File and Storage Technologies* (FAST), San Jose: Usenix Association, February 17–20, 2013.
37. Thomas Schwarz, Ignacio Corderí, Darrell D. E. Long and Jehan-François Pâris. "Simple, Exact Placement of Data in Containers," *Proceedings of International Conference on Computing, Networking and Communications* (ICNC), San Diego: IEEE, January 28–31, 2013.
38. Zhike Zhang, Deepavali Bhagwat, Witold Litwin, Darrell Long and Thomas Schwarz, S.J. "Improved Deduplication through Parallel Binning," *Proceedings of the International Performance Conference on Computers and Communication* (IPCCC), Austin: IEEE, December 1–3, 2012.
39. Jehan-François Pâris, Thomas Schwarz, S.J., Ahmed Amer and Darrell D. E. Long. "Highly Reliable Two-Dimensional RAID Arrays for Archival Storage," *Proceedings of the International Performance Conference on Computers and Communication* (IPCCC), Austin: IEEE, December 1–3, 2012.

40. Yulai Xie, Kiran-Kumar Muniswamy-Reddy, Dan Feng, Yan Li, Darrell D. E. Long, Zhipeng Tan and Lei Chen. "A Hybrid Approach for Efficient Provenance Storage," *Proceedings of International Conference on Information and Knowledge Management (CIKM)*, Maui: ACM, October 29–November 2, 2012.
41. Rekha Pitchumani, Andy Hospodor, Ahmed Amer, Yangwook Kang, Ethan L. Miller and Darrell Long. "Emulating a Shingled Write Disk," *Proceedings of the Twentieth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Washington, DC: IEEE, August 7–9, 2012.
42. Yan Li, Ethan L. Miller and Darrell D. E. Long. "Understanding Data Survivability in Archival Storage Systems," *Proceedings of the Fifth Annual International Systems and Storage Conference (SYSTOR)*, Haifa, Israel: ACM, June 2012.
43. Witold Litwin, Darrell Long and Thomas Schwarz. "Combining Chunk Boundary and Chunk Signature Calculations for Deduplication," *Proceedings of the Tenth International Information and Telecommunication Technologies Symposium (I2TS)*, Florianópolis, Brazil: IEEE, December 2011.
44. Yulai Xie, Kiran-Kumar Muniswamy-Reddy, Dan Feng, Darrell D. E. Long, Yangwook Kang, Zhongying Niu and Zhipeng Tan. "Design and Evaluation of Oasis: An Active Storage Framework based on T10 OSD Standard," *Proceedings of the Conference on Mass Storage Systems and Technologies (MSST)*, Denver, Colorado: IEEE, May 23–27, 2011.
45. Jehan-François Pâris, Ahmed Amer and Darrell Long. "Accelerated Chaining: A Better Way to Harness Peer Power in Video-on-Demand Applications," *Proceedings of the Twenty-sixth ACM Symposium on Applied Computing (SAC 2011)*, Taichung, Taiwan: ACM, March 2011.
46. Thomas J. E. Schwarz and Darrell Long. "Clasas: A Key-Store for the Cloud," *Proceedings of the Eighteenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Miami Beach: IEEE, August 17–19, 2010.
47. Jehan-François Pâris, Thomas J. E. Schwarz, Ahmed Amer and Darrell Long. "Improving Disk Array Reliability Through Expedited Scrubbing," *Proceedings of the Fifth IEEE International Conference on Networking, Architecture, and Storage (NAS 2010)*, Macau: IEEE, July 2010, pp. 119–125.
48. Aleatha Parker-Wood, Christina Strong, Ethan L. Miller and Darrell D. E. Long. "Security Aware Partitioning for Efficient File System Search," *Proceedings of the Conference on Mass Storage Systems and Technologies*, Incline Village, Nevada: IEEE, May 2010.
49. Ahmed Amer, Darrell D. E. Long, Ethan L. Miller, Jehan-François Pâris and Thomas J. E. Schwarz. "Design Issues for a Shingled Write Disk System," *Proceedings of the Conference on Mass Storage Systems and Technologies*, Incline Village, Nevada: IEEE, May 2010.
50. Jehan-François Pâris, Ahmed Amer, Darrell D. E. Long and Thomas J. E. Schwarz. "Evaluating the Impact of Irrecoverable Read Errors on Disk Array Reliability," *Proceedings of the IEEE Fifteenth Pacific Rim International Symposium on Dependable Computing (PRDC)*, Shanghai, China: IEEE, November 2009.
51. Jehan-François Pâris, Ahmed Amer and Darrell D. E. Long. "Using Storage Class Memories to Increase the Reliability of Two-dimensional RAID Arrays," *Proceedings of the Seventeenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, London: IEEE, September 21–23, 2009.
52. Avani Wildani, Thomas J. E. Schwarz, Ethan L. Miller and Darrell D. E. Long. "Protecting Against Rare Event Failures in Archival Systems," *Proceedings of the Seventeenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, London: IEEE, September 21–23, 2009.

53. Deepavali Bhagwat, Kave Eshghi, Darrell D. E. Long and Mark Lillibridge. "Extreme Binning: Scalable, Parallel Deduplication for Chunk-based File Backup," *Proceedings of the Seventeenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, London: IEEE, September 21–23, 2009, pp. 237–245.
54. Kevin M. Greenan, Darrell D. E. Long, Ethan L. Miller, Thomas J. E. Schwarz and Avani Wildani. "Building Flexible, Fault-Tolerant Flash-based Storage Systems," *Proceedings of Hot Topics in System Dependability (HotDep)*, Lisbon: IEEE, June 29, 2009.
55. Ahmed Amer, Jehan-François Pâris, Darrell D. E. Long and Thomas Schwarz. "Progressive Parity-Based Hardening of Data Stores," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Austin: IEEE, December 7–9, 2008, pp. 34–42.
56. Jehan-François Pâris, Thomas Schwarz, Darrell D. E. Long and Ahmed Amer. "When MTTLs Are Not Good Enough: Providing Better Estimates of Disk Array Reliability," *Proceedings of the Seventh International Information and Telecommunication Technologies Symposium (I2TS)*, Cuiabá, Brazil, Foz do Iguaçu, Paraná State, Brazil, December 3–5, 2008, pp. 190–195.
57. Kevin M. Greenan, Darrell D. E. Long, Ethan L. Miller, Thomas J. E. Schwarz, S.J. and Jay J. Wylie. "A Spin-Up Saved is Energy Earned: Achieving Power-Efficient, Erasure-Coded Storage," *Proceedings of Hot Topics in System Dependability (HotDep)*, San Diego: Usenix Association, December 7, 2008.
58. Mark W. Storer, Kevin Greenan, Darrell D. E. Long and Ethan L. Miller. "Secure Data Deduplication," *Proceedings of the Fourth International Workshop on Storage Security and Survivability (StorageSS)*, held in conjunction with the *ACM Computer and Communications Security Conference (CCS)*, Alexandria, Virginia: ACM, October 31, 2008.
59. Ahmed Amer, Darrell D. E. Long, Jehan-François Pâris and Thomas Schwarz. "Increased Reliability with SSPiRAL Data Layouts," *Proceedings of the Sixteenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Baltimore: IEEE, September 8–10, 2008, pp. 189–198.
60. Kevin Greenan, Ethan L. Miller, Thomas Schwarz and Darrell D. E. Long. "Disaster Recovery Codes: Increasing Reliability with Large-Stripe Error Correction Codes," *Proceedings of the Third International Workshop on Storage Security and Survivability (StorageSS)*, held in conjunction with the *ACM Computer and Communications Security Conference (CCS)*, Alexandria, Virginia: ACM, October 29, 2007.
61. Kristal T. Pollack, Darrell D. E. Long, Richard A. Golding, Ralph A. Becker-Szendy and Benjamin Reed. "Quota Enforcement for High-performance Distributed Storage Systems," *Proceedings of the Conference on Mass Storage Systems and Technologies*, San Diego: IEEE, September 2007, pp. 72–84.
62. Timothy Bisson, Scott A. Brandt and Darrell D. E. Long. "A Hybrid Disk-Aware Spin-Down Algorithm with I/O Subsystem Support," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, New Orleans: IEEE, April 11–13, 2007.
63. Jehan-François Pâris, Thomas J. Schwarz, S.J. and Darrell D. E. Long. "Self-Adaptive Two-Dimensional RAID Arrays," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, New Orleans: IEEE, April 11–13, 2007.
64. Jehan-François Pâris, Thomas J. Schwarz, S.J. and Darrell D. E. Long. "Protecting Data Against Early Disk Failures," *Proceedings of the Fifth International Information and Telecommunication Technologies Symposium (I2TS)*, Cuiabá, Brazil, December 2006, pp. 113–119.
65. Jehan-François Pâris, Thomas J. Schwarz, S.J. and Darrell D. E. Long. "Self-Adaptive Disk Arrays," *Proceedings of the Eighth International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS)*, Springer, Lecture Notes in Computer Science: Dallas, November 2006, pp. 469–483.

66. Sage A. Weil, Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long and Carlos Maltzahn. "Ceph: A Scalable, High-Performance Distributed File System," *Proceedings of the Seventh Symposium on Operating Systems Design and Implementation (OSDI)*, Seattle: Usenix Association, November 2006, pp. 307–320.
67. Timothy Bisson, Scott A. Brandt and Darrell D. E. Long. "NVCash: Increasing the Effectiveness of Disk Spin-down Algorithms with Caching," *Proceedings of the Fourteenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Monterey: IEEE, September 2006, pp. 422–432.
68. Jeffrey P. Rybczynski, Darrell D. E. Long and Ahmed Amer. "Adapting Predictions and Workloads for Power Management," *Proceedings of the Fourteenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Monterey: IEEE, September 2006, pp. 3–12.
69. Deepavali Bhagwat, Kristal Pollack, Darrell D. E. Long, Ethan L. Miller, Thomas Schwarz, S.J. and Jehan-François Pâris. "Providing High Reliability in a Minimum Redundancy Archival Storage System," *Proceedings of the Fourteenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Monterey: IEEE, September 2006, pp. 412–421.
70. Lawrence L. You, Kristal T. Pollack and Darrell D. E. Long. "Deep Store: an Archival Storage System Architecture," *Proceedings of the Twenty-first International Conference on Data Engineering (ICDE)*, Tokyo: IEEE, April 2005, pp. 804–815.
71. Qin Xin, Ethan L. Miller, Thomas J. E. Schwarz, and Darrell D. E. Long. "Impact of Failure on Interconnection Networks in Large Storage Systems," *Proceedings of the Twenty-second Conference on Mass Storage Systems and Technologies*, Monterey: IEEE, April 2005, pp. 189–196.
72. Feng Wang, Bo Hong, Scott A. Brandt and Darrell D. E. Long. "Using MEMS-based Storage to Boost Disk Performance," *Proceedings of the Twenty-second Conference on Mass Storage Systems and Technologies*, Monterey: IEEE, April 2005, pp. 202–209.
73. Thomas J. E. Schwarz, Qin Xin, Ethan L. Miller, Darrell D. E. Long, Andrew Hospodor and Spencer Ng. "Disk Scrubbing in Large Archival Storage Systems," *Proceedings of the Twelfth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Volendam, The Netherlands: IEEE, October 2004, pp. 409–418.
74. Karl Brandt, Darrell D. E. Long and Ahmed Amer. "Predicting When Not To Predict," *Proceedings of the Twelfth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Volendam, The Netherlands: IEEE, October 2004, pp. 419–426.
75. Bo Hong, Thomas J. E. Schwarz, Scott A. Brandt and Darrell D. E. Long. "Reliability of MEMS-Based Storage Enclosures," *Proceedings of the Twelfth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Volendam, The Netherlands: IEEE, October 2004, pp. 571–579.
76. Bo Hong, Demyan Plantenberg, Darrell D. E. Long and Miriam Sivan-Zimet. "Duplicate Data Elimination in a SAN File System," *Proceedings of the Twenty-first Symposium on Mass Storage Systems (MSS)*, Goddard, Maryland: NASA, April 2004, pp. 301–314.
77. Purvi Shah, Jehan-François Pâris, Ahmed Amer and Darrell D. E. Long. "Identifying Stable File Access Patterns," *Proceedings of the Twenty-first Symposium on Mass Storage Systems (MSS)*, Goddard, Maryland: NASA, April 2004, pp. 159–163.
78. Feng Wang, Qin Xin, Bo Hong, Scott A. Brandt, Ethan Miller, Darrell Long and Tyce T. McLarty. "File System Workload Analysis For Large Scale Scientific Computing Applications," *Proceedings of the Twenty-first Symposium on Mass Storage Systems (MSS)*, Goddard, Maryland: NASA, April 2004, pp. 139–152.

79. Feng Wang, Scott A. Brandt, Ethan L. Miller and Darrell D. E. Long. "OBFS: A File System for Object-Based Storage Devices," *Proceedings of the Twenty-first Symposium on Mass Storage Systems (MSS)*, Goddard, Maryland: NASA, April 2004, pp. 283–300.
80. Ismail Ari, Bo Hong, Ethan L. Miller, Scott A. Brandt and Darrell D. E. Long. "Managing Flash Crowds on the Internet," *Proceedings of the Eleventh International Symposium on Modeling, Analysis, and Simulation in Computer and Telecommunication Systems (MASCOTS)*, Orlando: IEEE, October 2003, pp. 246–249.
81. Bo Hong, Scott A. Brandt, Darrell D. E. Long, Ethan L. Miller, Karen A. Glocer and Zachary N. J. Peterson. "Zone-Based Shortest Positioning Time First Scheduling for MEMS-Based Storage Devices," *Proceedings of the Eleventh International Symposium on Modeling, Analysis, and Simulation in Computer and Telecommunication Systems (MASCOTS)*, Orlando: IEEE, October 2003, pp. 104–113.
82. Karthik Thirumalai, Jehan-François Pâris and Darrell D. E. Long. "Tabbycat: An Inexpensive Scalable Server for Video-on-Demand," *Proceedings of the IEEE 2003 International Conference on Communications (ICC)*, Anchorage, Alaska: IEEE, May 2003, pp. 896–900.
83. Ahmed Amer, Alison Luo, Newton Der, Darrell D. E. Long and Alex Pang. "Visualizing Cache Effects on I/O Workload Predictability," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, April 2003, pp. 417–424.
84. Jehan-François Pâris and Darrell D. E. Long. "A Proactive Implementation of Interactive Video-on-Demand," *Proceedings of the International Performance, Computing, and Communications Conference (IPCCC)*, Phoenix: IEEE, April 2003, pp. 425–431.
85. Qin Xin, Ethan L. Miller, Scott A. Brandt, Darrell D. E. Long, Thomas Schwarz and Witold Litwin. "Reliability Mechanisms for Very Large Storage Systems," *Proceedings of the Twentieth Symposium on Mass Storage Systems (MSS)*, San Diego: IEEE, April 2003, pp. 146–156.
86. Gary A. S. Whittle, Jehan-François Pâris, Ahmed Amer, Darrell D. E. Long and Randal Burns. "Using Multiple Predictors to Improve the Accuracy of File Access Predictions," *Proceedings of the Twentieth Symposium on Mass Storage Systems (MSS)*, San Diego: IEEE, April 2003, pp. 230–240.
87. Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long and Lan Xue. "Efficient Metadata Management in a Large Distributed Storage System," *Proceedings of the Twentieth Symposium on Mass Storage Systems (MSS)*, San Diego: IEEE, April 2003, pp. 290–298.
88. Jehan-François Pâris and Darrell D. E. Long. "A Variable Bandwidth Broadcasting Protocol For Video-on-Demand," *Proceedings of the 2003 SPIE Multimedia Computing and Networking Conference (MMCN)*, San Jose: SPIE, January 2003, pp. 209–219.
89. Ying Lin, Scott A. Brandt, Darrell D. E. Long and Ethan L. Miller. "Power Conservation Strategies for MEMS-based Storage Devices," *Proceedings of the Tenth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Fort Worth: IEEE, October 2002, pp. 53–62.
90. Ahmed Amer, Darrell D. E. Long and Randal C. Burns. "Group-Based Management of Distributed File Caches," *Proceedings of the Twenty-second International Conference on Distributed Computing Systems (ICDCS)*, Vienna: IEEE, July 2002, pp. 525–534.
91. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Increasing Predictive Accuracy by Prefetching Multiple Program and User Specific Files," *Proceedings of the Sixteenth Annual International Symposium on High Performance Computing Systems and Applications (HPCS)*, Moncton, New Brunswick, Canada: IEEE, June 2002, pp. 12–19.
92. Alison Luo, Ahmed Amer, Scott Speidel, Darrell D. E. Long and Alex Pang. "Visualizing I/O Predictability," *Proceedings of the First International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, Padova, Italy: IEEE, June 2002, pp. 202–207.

93. Ahmed Amer, Darrell D. E. Long, Jehan-François Pâris and Randal C. Burns. "File Access Prediction with Adjustable Accuracy," *Proceedings of the International Performance Conference on Computers and Communication* (IPCCC), Phoenix: IEEE, April 2002, pp. 131–140.
94. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Increasing Predictive Accuracy through Limited Prefetching," *Proceedings of Communications Networks and Distributed Systems Modeling and Simulation* (CNDS), San Antonio: Society for Computer Simulation, January 2002, pp. 131–138.
95. Ethan L. Miller, Darrell D. E. Long, William Freeman and Benjamin C. Reed. "Strong Security for Network-Attached Storage," *Proceedings of the Conference on File and Storage Technologies* (FAST), Monterey: Usenix Association, January 2002, pp. 1–13.
96. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Using Program and User Information to Improve File Prediction Performance," *Proceedings of the International Symposium on Performance Analysis of Systems and Software* (ISPASS), Tucson: IEEE, November 2001, pp. 111–119.
97. Ahmed Amer and Darrell D. E. Long. "Aggregating Caches: A Mechanism for Implicit File Prefetching," *Proceedings of the Ninth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems* (MASCOTS), Cincinnati: IEEE, August 2001, pp. 293–301.
98. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Performing File Prediction with a Program-Based Successor Model," *Proceedings of the Ninth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems* (MASCOTS), Cincinnati: IEEE, August 2001, pp. 193–202.
99. Thomas M. Kroeger and Darrell D. E. Long. "Design and Implementation of a Predictive File Prefetching Algorithm," *Proceedings of Usenix Technical Conference*, Boston: Usenix Association, June 2001, pp. 105–118.
100. Ethan L. Miller, Scott A. Brandt and Darrell D. E. Long. "HeRMES: High-Performance Reliable MRAM-Enabled Storage," *Proceedings of the Eighth Workshop on Hot Topics in Operating Systems* (HotOS-VIII), Elmau, Germany: IEEE, May 2001, pp. 95–99.
101. Ahmed Amer and Darrell D. E. Long. "Noah: Low-cost File Access Prediction Through Pairs," *Proceedings of the International Performance Conference on Computers and Communication* (IPCCC), Phoenix: IEEE, April 2001, pp. 27–33.
102. Ethan Miller, Darrell Long, William Freeman and Benjamin Reed. "Strong Security for Distributed File Systems," *Proceedings of the International Performance Conference on Computers and Communication* (IPCCC), Phoenix: IEEE, April 2001, pp. 34–40.
103. Scott R. Carter, Jehan-François Pâris, Saurabh Mohan and Darrell D. E. Long. "A Dynamic Heuristic Broadcasting Protocol for Video-on-Demand," *Proceedings of the Twenty-first International Conference on Distributed Computing Systems* (ICDCS), Mesa, Arizona: IEEE, April 2001, pp. 657–664.
104. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "An Analytical Study of Opportunistic Lease Renewal," *Proceedings of the Twenty-first International Conference on Distributed Computing Systems* (ICDCS), Mesa, Arizona: IEEE, April 2001, pp. 146–153.
105. Steven W. Carter, Darrell D. E. Long and Jehan-François Pâris. "Efficient Implementation of Interactive Video-on-Demand," *Proceedings of the Eighth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), San Francisco: IEEE, August–September 2000, pp. 172–179.
106. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "Consistency and Locking for Distributing Updates to Web Servers Using a File System," *Proceedings of Performance and Architecture of Web Servers* (PAWS), Santa Clara: ACM, June, 2000 and *Performance Evaluation Review*, vol. 28, no. 2, September 2000.

107. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "Safe Caching in a Distributed File System for Network Attached Storage," *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*, Cancun: IEEE, May 2000, pp. 155–162.
108. Jehan-François Pâris, Theodore R. Haining and Darrell D. E. Long. "A Stack Model Based Replacement Policy for a Non-Volatile Write Cache," *Proceedings of the Eighth NASA Goddard Conference on Mass Storage Systems and Technologies*, College Park: IEEE, March 2000, pp. 217–223.
109. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "Semi-Preemptible Locks for a Distributed File System," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, February 2000, pp. 397–404.
110. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "A Reactive Broadcasting Protocol for Video on Demand," *Proceedings of the Multimedia Computing and Networking Conference (MMCN)*, San Jose: SPIE, January 2000, pp. 216–223.
111. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "Combining Pay-per-View and Video-on-Demand Services," *Proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, College Park: IEEE, October 1999, pp. 270–276.
112. Jehan-François Pâris, Darrell D. E. Long and Patrick E. Mantey. "A Zero-delay Broadcasting Protocol for Video on Demand," *Proceedings of the Seventh ACM International Multimedia Conference*, Orlando: ACM, October 1999, pp. 189–197.
113. Benjamin C. Reed, Darrell D. E. Long, Edward G. Chron and Randal C. Burns. "Authenticating Network Attached Storage," *Proceedings of the Symposium on High Performance Interconnects (Hot Interconnects VII)*, Stanford: IEEE, August 1999, pp. 49–57.
114. Thomas M. Kroeger and Darrell D. E. Long. "The Case for Efficient File Access Pattern Modeling," *Proceedings of the Seventh Workshop on Hot Topics in Operating Systems (HotOS-VII)*, Rio Rico, Arizona: IEEE, March 1999, pp. 14–19.
115. Theodore R. Haining and Darrell D. E. Long. "Management Policies for Non-volatile Write Caches," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, February 1999, pp. 321–328.
116. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "A Hybrid Broadcasting Protocol for Video on Demand," *Proceedings of the Multimedia Computing and Networking Conference (MMCN)*, San Jose: SPIE, January 1999, pp. 317–326.
117. Timothy Gibson, Ethan L. Miller and Darrell D. E. Long. "Long-term File Activity and Inter-reference Patterns," *Proceedings of the Computer Measurement Group Conference*, Anaheim: CMG, December 1998, pp. 976–987.
118. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "A Low Bandwidth Broadcasting Protocol for Video on Demand," *Proceedings of the International Conference on Computer Communication and Networks (ICCCN)*, Lafayette: IEEE, October 1998, pp. 690–697.
119. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "Efficient Broadcasting Protocols for Video on Demand," *Proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Montreal: IEEE, July 1998, pp. 127–130.
120. Randal C. Burns and Darrell D. E. Long. "In-Place Reconstruction of Delta Compressed Files," *Proceedings of Principles of Distributed Computing (PODC)*, Puerto Vallarta: ACM, June 1998, pp. 267–275.

121. Thomas M. Kroeger, Darrell D. E. Long and Jeffrey C. Mogul. "Exploring the Bounds of Web Latency Reduction from Caching and Prefetching," *Proceedings of the Symposium on Internet Technologies and Systems*, Monterey: Usenix Association, 1997, pp. 13–22.
122. Randal C. Burns and Darrell D. E. Long. "Efficient Distributed Back-up with Delta Compression," *Proceedings of I/O in Parallel and Distributed Systems (IOPADS)*, San Jose: ACM, November 1997, pp. 27–36.
123. Steven W. Carter and Darrell D. E. Long. "Improving Video-on-Demand Server Efficiency Through Stream Tapping," *Proceedings of the Sixth International Conference on Computer Communications and Networks (ICCCN)*, Las Vegas: IEEE, September 1997, pp. 200–207.
124. Darrell D. E. Long and Jehan-François Pâris. "Voting Without Version Numbers," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, February 1997, pp. 139–145.
125. Randal C. Burns and Darrell D. E. Long. "A Linear Time, Constant Space Differencing Algorithm," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, February 1997, pp. 429–436.
126. David P. Helmbold, Darrell D. E. Long and Bruce Sherrod. "A Dynamic Disk Spin-down Technique for Mobile Computing," *Proceedings of the Second Annual International Conference on Mobile Computing and Networking (MobiCom)*, Rye, New York: ACM, November 1996, pp. 130–142.
127. Darrell D. E. Long and Jehan-François Pâris. "A Leaner, More Efficient, Available Copy Protocol," *Proceedings of the Symposium on Parallel and Distributed Processing (SPDP)*, New Orleans: IEEE, October 1996, pp. 400–407.
128. Darrell D. E. Long, Patrick E. Mantey, Eric C. Rosen, Craig M. Wittenbrink and Bruce Gritton. "REINAS: A Real-time System for Managing Environmental Data," *Proceedings of the Eighth Software Engineering and Knowledge Engineering Conference*, Lake Tahoe: SEKE, June 1996, pp. 293–300.
129. Thomas M. Kroeger and Darrell D. E. Long. "Predicting File-System Actions from Prior Events," *Proceedings of 1996 Usenix Winter Technical Conference*, San Diego: Usenix Association, January 1996, pp. 319–328.
130. Darrell Long, Andrew Muir and Richard Golding. "A Longitudinal Study of Internet Host Reliability," *Proceedings of the Fourteenth Symposium on Reliable Distributed Systems (SRDS)*, Bad Neuenahr, Germany: IEEE, September 1995, pp. 2–9.
131. Chane L. Fullmer, Darrell D. E. Long and Luis-Felipe Cabrera. "Adding Adaptive Flow Control to Swift/RAID," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, March 1995, pp. 290–296.
132. Darrell D. E. Long, Patrick E. Mantey, Theodore R. Haining, Bruce R. Montague, Eric C. Rosen. "An Architecture Supporting Real-Time and Retrospective Environmental Data Management," *Proceedings of the First International Conference for Applications of Databases (ADB)*, June 1994.
133. Cheng Tang, Shree Murthy, and Darrell D. E. Long. "Performance Guarantees on ATM Networks," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, April 1994, pp. 111–118.
134. Richard A. Golding, Darrell D. E. Long and John Wilkes. "The *refdbms* Distributed Bibliographic Database System," *Proceedings of the 1994 Usenix Winter Technical Conference*, San Francisco: Usenix Association, January 1994, pp. 47–62.
135. Richard A. Golding and Darrell D. E. Long. "Using an Object-oriented Framework to Construct Wide-area Group Communication Mechanisms," *Proceedings of the International Symposium on Applied Computing*, Monterrey, Mexico, October 1993, pp. 65–74.

136. Darrell D. E. Long, Carol Osterbrock and Luis-Felipe Cabrera. "Providing Performance Guarantees in an FDDI Network," *Proceedings of the Thirteenth International Conference on Distributed Computing Systems (ICDCS)*, Pittsburgh: IEEE, May 1993, pp. 328–336.
137. Darrell D. E. Long and Madhukar N. Thakur. "Scheduling Real-Time Disk Transfers for Continuous Media Applications," *Proceedings of the Twelfth Symposium on Mass Storage Systems (MSS)* Monterey: IEEE, April 1993, pp. 227–232.
138. Richard A. Golding and Darrell D. E. Long. "Simulation Modeling of Weak-Consistency Protocols," *Proceedings of the First International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, San Diego: Society for Computer Simulation, January 1993, pp. 233–238.
139. Perry K. Sloope, Jehan-François Pâris and Darrell D. E. Long. "A Simulation Study of Replication Control Protocols Using Volatile Witnesses," *Proceedings of the Twenty-fifth Annual Simulation Symposium*, Orlando: IEEE, April 1992, pp. 108–117.
140. Jehan-François Pâris and Darrell D. E. Long. "Protecting Replicated Objects Against Media Failures," *Proceedings of the Workshop on Research Issues on Transaction and Query Processing*, Phoenix: IEEE, February 1992, pp. 109–115.
141. Richard A. Golding and Darrell D. E. Long. "Quorum-oriented Multicast Protocols for Data Replication," *Proceedings of the Eighth International Conference on Data Engineering (ICDE)*, Phoenix: IEEE, February 1992, pp. 490–497.
142. Luis-Felipe Cabrera and Darrell D. E. Long. "Swift: A Distributed Storage Architecture for Large Objects," *Proceedings of the Eleventh Symposium on Mass Storage Systems (MSS)*, Monterey: IEEE, October 1991, pp. 123–128.
143. Darrell D. E. Long, John L. Carroll and C. J. Park. "A Study of the Reliability of Internet Sites," *Proceedings of the Tenth Symposium on Reliable Distributed Systems (SRDS)*, Pisa: IEEE, September 1991, pp. 177–186.
144. Darrell D. E. Long and Luis-Felipe Cabrera. "Exploiting Multiple I/O Streams to Provide High Data-Rates," *1991 Summer Usenix Technical Conference*, Nashville: Usenix Association, June 1991, pp. 31–48.
145. Jehan-François Pâris and Darrell D. E. Long. "Voting with Regenerable Volatile Witnesses," *Proceedings of the Seventh International Conference on Data Engineering (ICDE)*, Kobe, Japan: IEEE, April 1991, pp. 112–119.
146. Jehan-François Pâris and Darrell D. E. Long. "Resilient Memory-Resident Data Objects," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, March 1991, pp. 145–151.
147. Darrell D. E. Long, John L. Carroll and Kris Stewart. "The Reliability of Regeneration-Based Replica Control Protocols," *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, Newport Beach: IEEE, June 1989, pp. 465–473.
148. John L. Carroll and Darrell D. E. Long. "The Effect of Failure and Repair Distributions on Consistency Protocols for Replicated Data Objects," *Proceedings of the Twenty-second Annual Simulation Symposium*, Tampa: IEEE, March 1989, pp. 47–60.
149. Darrell D. E. Long, Jehan-François Pâris and John L. Carroll. "Reliability of Replicated Data Objects," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, March 1989, pp. 402–406.
150. Darrell D. E. Long and Jehan-François Pâris. "Regeneration Protocols for Replicated Objects," *Proceedings of the International Conference on Data Engineering (ICDE)*, Los Angeles: IEEE, February 1989, pp. 538–545.

151. Darrell D. E. Long and Jehan-François Pâris. "A Realistic Evaluation of Optimistic Dynamic Voting," *Proceedings of the Seventh Symposium on Reliable Distributed Systems (SRDS)*, Columbus: IEEE, October 1988, pp. 129–137.
152. Jehan-François Pâris, Darrell D. E. Long and Alexander Glockner. "A Realistic Evaluation of Consistency Algorithms for Replicated Files," *Proceedings of the Twenty-first Annual Simulation Symposium*, Tampa: IEEE, March 1988, pp. 121–130.
153. Jehan-François Pâris and Darrell D. E. Long. "Efficient Dynamic Voting Algorithms," *Proceedings of the International Conference on Data Engineering (ICDE)*, Los Angeles: IEEE, February 1988, pp. 268–275.
154. John L. Carroll, Darrell D. E. Long and Jehan-François Pâris. "Block-Level Consistency of Replicated Files," *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, Berlin: IEEE, September 1987, pp. 146–153.
155. Darrell D. E. Long and Jehan-François Pâris. "On Improving the Availability of Replicated Files," *Proceedings of the Symposium on Reliability in Distributed Software and Database Systems (SRDSDS)*, Williamsburg: IEEE, March 1987, pp. 77–83.

Peer Reviewed Short Papers

1. Oceane Bel, Sinjoni Mukhopadhyay, Nathan Tallent, Faisal Nawab and Darrell Long. "WinnnowML: Stable Feature Selection for Maximizing Prediction Accuracy of Time-based System Modeling," *IEEE International Conference on Big Data (Big Data)*, December 15–18, 2021, *to appear*.
2. Oceane Bel, Kenneth Chang, Nathan R. Tallent, Dirk Duellmann, Ethan L. Miller, Faisal Nawab and Darrell D. E. Long. "Geomancy: Automated Performance Enhancement through Data Layout Optimization," *Proceedings of the International Symposium on Performance Analysis of Systems and Software (ISPASS)*, IEEE: Virtual, August 23–26, 2020.
3. Austen Barker, Staunton Sample, Yash Gupta, Anastasia McTaggart, Ethan L. Miller and Darrell D. E. Long. "Artifice: A Deniable Steganographic File System," *Proceedings of Ninth Usenix Workshop on Free and Open Communications on the Internet*, (FOCI 2019), Usenix Association: Santa Clara, August 13, 2019.
4. Jehan-François Pâris, Thomas Schwarz, S. J. and Darrell D. E. Long. "Improving Disk Array Reliability Through Faster Repairs," *Proceedings of the Thirty-fifth International Performance of Computers and Communication Conference (IPCCC 2016)*, Las Vegas, NV: IEEE, December 2016.
5. Daniel Bittman, D.J. Capelis and Darrell Long. "Introducing SeaOS," *Proceedings of the 2014 International Conference on Information Science and Applications (ICISA)*, Seoul, Korea: IEEE, May 6–9, 2014, pp. 1–3.
6. Ranjana Rajendran, Ethan L. Miller and Darrell D. E. Long. "Horus: Fine-Grained Encryption-Based Security for High Performance Petascale Storage," *Proceedings of the Sixth Petascale Data Storage Workshop (PDSW)*, Seattle: ACM, November 13, 2011.
7. Stephanie N. Jones, Christina R. Strong, Aleatha Parker-Wood, Alexandra Holloway and Darrell D. E. Long. "Easing the Burdens of HPC File Management," *Proceedings of the Fifth Petascale Data Storage Workshop (PDSW)*, Seattle: ACM, November 13, 2011.
8. Yulai Xie, Kiran-Kumar Muniswamy-Reddy, Darrell D. E. Long, Ahmed Amer, Dan Feng and Zhipeng Tan. "Compressing Provenance Graphs," *Proceedings of the Workshop on the Theory and Practice of Provenance (TaPP)*, Crete: Usenix Association, June 20–21, 2011.
9. Stephanie N. Jones, Christina R. Strong, Darrell D. E. Long and Ethan L. Miller. "Tracking Emigrant File Data via Transient Provenance," *Proceedings of the Workshop on the Theory and Practice of Provenance (TaPP)*, Crete: Usenix Association, June 20–21, 2011.

10. Ignacio Corderi, Thomas Schwarz, Ahmed Amer, Darrell D. E. Long and Jehan-François Pâris. "Self-Adjusting Two-Failure Tolerant Disk Arrays," *Proceedings of the Fifth Petascale Data Storage Workshop (PDSW)*, New Orleans: ACM, November 15, 2010.
11. Sara Chaarawi, Jehan-François Pâris, Ahmed Amer, Thomas Schwarz and Darrell D. E. Long. "Using a Shared Storage Class Memory Device to Improve the Reliability of RAID Arrays," *Proceedings of the Fifth Petascale Data Storage Workshop (PDSW)*, New Orleans: ACM, November 15, 2010.
12. D. J. Capelis and Darrell D. E. Long. "Fived: A Service-Based Architecture Implementation to Innovate at the Endpoints," *Proceedings of SIGCOMM 2010*, New Dehli, India: ACM, September 2010, pp. 419–420.
13. Ian F. Adams, Darrell D. E. Long and Ethan L. Miller. "Maximizing Efficiency By Trading Storage for Computation," *Proceedings of the Workshop on Hot Topics in Cloud Computing (Hot Cloud)*, San Diego: Usenix Association, June 2009.
14. Mark W. Storer, Kevin M. Greenan, Ian F. Adams Ethan L. Miller, Darrell D. E. Long and Kaladhar Voruganti. "Logan: Automatic Management for Evolvable, Large-Scale, Archival Storage," *Proceedings of the 2008 Petascale Data Storage Workshop (PDSW)*, IEEE, November 2008, pp. 1–8.
15. Neerja Bhatnagar, Kevin M. Greenan, Rosie Wacha, Ethan L. Miller and Darrell D. E. Long. "Energy-Reliability Tradeoffs in Sensor Network Storage," *Proceedings of Fifth Workshop on Embedded Networked Sensors (HotEm-Nets)*, Charlottesville, Virginia: ACM, June 2–3, 2008.
16. Jehan-François Pâris, Darrell D. E. Long and Thomas J. E. Schwarz. "Ensuring Data Survival in Solid-State Storage Devices," *Proceedings of the First International Workshop on Storage and I/O Virtualization, Performance, Energy, Evaluation and Dependability (SPEED)*, Salt Lake City: IEEE, February 2008.
17. Jehan-François Pâris and Darrell D. E. Long. "Using Device Diversity to Protect Data against Batch-Related Disk Failures," *Proceedings of the Second International Workshop on Storage Security and Survivability (StorageSS)*, held in conjunction with the *ACM Computer and Communications Security Conference (CCS)*, Alexandria, Virginia: ACM, October 30, 2006.
18. Jehan-François Pâris and Darrell D. E. Long. "An Analytic Study of Stream Tapping Protocols," *Proceedings of the 2006 IEEE International Conference on Multimedia (ICME)*, Toronto: IEEE, July 2006, pp. 1237–1240.
19. Jeffrey P. Rybczynski, Darrell D. E. Long and Ahmed Amer. "Expecting the Unexpected: Adaptation for Predictive Energy Conservation," *Proceedings of the International Workshop on Storage Security and Survivability (StorageSS)*, held in conjunction with the *ACM Computer and Communications Security Conference (CCS)*, Alexandria, Virginia: ACM, November 2005, pp. 130–134.
20. Jehan-François Pâris, Ahmed Amer and Darrell D. E. Long. "A Stochastic Approach to File Access Prediction," *Proceedings of the International Workshop on Storage Network Architecture and Parallel I/O (SNAPI)*, New Orleans: IEEE, September 2003.
21. Ismail Ari, Ahmed Amer, Robert Gramacy, Ethan Miller, Scott Brandt and Darrell Long. "Who is more Adaptive? ACME: Adaptive Caching Using Multiple Experts," *Proceedings of the 2002 Workshop on Distributed Data and Structures (WDAS)*, Paris, France: Carleton Scientific, March 2002.
22. Jehan-François Pâris and Darrell D. E. Long. "The Case for Aggressive Partial Preloading in Video-on-Demand Broadcasting Protocols," *Proceedings of the 2001 IEEE International Conference on Multimedia (ICME)*, Tokyo, Japan: IEEE, August 22–25, 2001, pp. 113–116.
23. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Caching Files with a Program-based Last n Successors Model," *Proceedings of the Workshop on Caching, Coherence and Consistency (WC3)*, Sorrento, Italy: ACM, June 2001.

24. Ahmed Amer and Darrell D. E. Long. "Adverse Filtering Effects and the Resilience of Aggregating Caches," *Proceedings of the Workshop on Caching, Coherence and Consistency (WC3)*, Sorrento, Italy: ACM, June 2001.
25. Tsozen Yeh, Darrell D. E. Long and Scott A. Brandt. "Conserving Battery Energy through Making Fewer Incorrect File Predictions," *Proceedings of the IEEE Workshop on Power Management for Real-Time and Embedded Systems* (at the IEEE Real-Time Technology and Applications Symposium), Taipei, Taiwan: IEEE, May 2001, pp. 30–36.
26. Ahmed Amer and Darrell D. E. Long. "Dynamic Relationships and the Persistence of Pairings," *Proceedings of the International Workshop on Wireless Networks and Mobile Computing (WNMC)*, Mesa, Arizona: IEEE, April 2001, pp. 502–507.
27. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "A Universal Distribution Protocol for Video-on-Demand," *Proceedings of the 2000 IEEE International Conference on Multimedia (ICME)*, New York: IEEE, July 2000, vol. 1, pp. 49–52.
28. Jehan-François Pâris and Darrell D. E. Long. "Limiting the Client Bandwidth of Broadcasting Protocols for Videos on Demand," *Proceedings of Euromedia 2000*, Antwerp: Society for Computer Simulation, May 2000, pp. 107–111.
29. Jehan-François Pâris, Steven W. Carter and Darrell D. E. Long. "Bandwidth Allocation Issues in Near Video on Demand Services," *Proceedings of the Third Annual Multimedia Technology and Applications Conference (MTAC)*, Anaheim: IEEE, September 1998, pp. 196–200.
30. Darrell D. E. Long. "A Replicated Monitoring Tool," *Proceedings of the Second Workshop on the Management of Replicated Data (WMRD)*, Monterey: IEEE, November 1992, pp. 96–99.
31. Darrell D. E. Long. "Analysis of Replication Control Protocols," *Proceedings of the First Workshop on the Management of Replicated Data (WMRD)*, Houston: IEEE, November 1990, pp. 117–122.
32. Jehan-François Pâris, Darrell D. E. Long and Walter A. Burkhard. "The Management of Consistency in Fault-Tolerant File Systems," *Proceedings of the Workshop on Fault Tolerance in Parallel and Distributed Computing*, San Diego: IEEE, December 1987, pp. 58–59.
33. Darrell D. E. Long. "Optimistic Algorithms for Replicated Data Management," *Proceedings of the Second Workshop on Large-Grained Parallelism*, Carnegie-Mellon University, November 1987, pp. 58–59.

Invited Articles

1. David Pease and Darrell D.E. Long. "Future File Systems," *Proceedings of Computing with Massive and Persistent Data (CMPD)*, Baltimore: IEEE, September 22, 2008.
2. Randal Burns, Larry Stockmeyer and Darrell D.E. Long. "Experimentally Evaluating In-Place Delta Reconstruction," *Proceedings of the NASA and IEEE Mass Storage Conference*, College Park: IEEE, April 2002, pp. 137–151.
3. Darrell D. E. Long. "A Technique for Managing Mirrored Disks," *Proceedings of the International Performance Conference on Computers and Communication (IPCCC)*, Phoenix: IEEE, April 2001, pp. 272–277.
4. Darrell D. E. Long, Patrick E. Mantey, Craig M. Wittenbrink, Theodore R. Haining and Bruce R. Montague. "REINAS: the Real-Time Environmental Information Network and Analysis System," *Proceedings of the IEEE Computer Society CompCon*, San Francisco: IEEE, March 1995, pp. 482–487.

Miscellanea

1. Daniel Bittman, Peter Alvaro, Darrell D. E. Long and Ethan Miller. "The Flipside: A Bit Flip Saved Is Power and Lifetime Earned," *login: The Usenix Magazine*, 2019, vol. 44, no. 2.
2. Ahmed Amer, Darrell D. E. Long, Ethan L. Miller, Jehan-Fraçois Pâris and Thomas Schwarz. "Data Management and Layout for Shingled Magnetic Recording," *IEEE International Magnetism Conference* (InterMag 2011), Taipei, Taiwan: IEEE, April 25–29, 2011.
3. K. Gopinath, Jon Elerath and Darrell Long. "Reliability Modelling of Disk Subsystems with Probabilistic Model Checking," *Dagstuhl Seminar on Quantitative and Qualitative Analysis of Network Protocols*, January 2010.
4. António M. Baptista, Michael Wilkin, Phillip Pearson, Paul Turner, Cole M^cCandlish, Phillip Barrett, Salil Das, Wendy Sommerfield, Ming Qi, Neetu Nangia, David Jay, Darrell Long, Calton Pu, John Hunt, Zhaoqing Yang, Edward Myers, Jeff Darland and Anna Farrenkopf, "Towards a Multipurpose Forecast Systems for the Columbia River Estuary," *Proceedings of the Ocean Community Conference*, Baltimore: Marine Technology Society, November 1998.
5. Benjamin Reed and Darrell D. E. Long, "Analysis of Caching Algorithms for Distributed File Systems," *Operating Systems Review*, July 1996, vol. 30, no. 3, pp. 12–21.
6. Daniel M. Fernandez, Patrick E. Mantey, Darrell D. E. Long, Eric Rosen and Craig M. Wittenbrink. "REINAS: Real-Time Environmental Information Network," *Sea Technology*, May 1996, vol. 37, no. 5, pp. 47–53.
7. Darrell D. E. Long. "A Note On Bit-Mapped Free Sector Management," *Operating Systems Review*, April 1993, vol. 27, no. 2, pp. 7–9.

Patents

1. Darrell D. E. Long and Thomas M. Kroeger. "Predictive Event Tracking Method," U.S. Patent no. 5,889,993, March 1999.
2. Luis Felipe Cabrera and Darrell D. E. Long. "Method and Apparatus for Establishing and Maintaining the Status of Membership Sets used in Mirrored Read and Write Input/Output without Logging," U.S. Patent no. 5,917,998, June 1999.
3. Randal C. Burns, Edward G. Chron, Darrell D. E. Long and Benjamin C. Reed. "Secure Array of Remotely Encrypted Storage Devices," U.S. Patent no. 5,931,947, August 1999.
4. Randal C. Burns and Darrell D. E. Long. "A Method for Generating and Reconstructing In-place Delta Files," U.S. Patent no. 6,018,747, January 2000.
5. Edward G. Chron, Darrell D. E. Long, Randal C. Burns and Benjamin Reed. "Decentralized Remotely Encrypted File System," U.S. Patent no. 6,405,315, June 2002.
6. Randal C. Burns, Robert M. Rees and Darrell D. E. Long. "Data Placement and Allocation using Virtual Contiguity," U.S. Patent no. 6,651,147, November 2003.
7. Randal C. Burns, Robert M. Rees, Darrell D. E. Long and Atul Goel. "Lease-Based Safety Protocol for Distributed System with Multiple Networks," U.S. Patent no. 6,775,703, August 2004.
8. Randal C. Burns and Darrell D. E. Long. "System and Method for Managing Authentication and Coherency in a Storage Area Network," U.S. Patent no. 6,792,424, September 2004.
9. Randal C. Burns, Darrell D. E. Long and Robert M. Rees. "System and Method for Restoring a File System from Backups in the Presence of Deletions," U.S. Patent no. 6,938,056, August 2005.

10. Rajagopal Ananthanarayanan, Ralph Becker-Szendy, Randal C. Burns, Darrell D. E. Long, Jujjuri Venkateswararao, David M. Wolf and Jason C. Young. "Method, System and Computer Program Product for Implementing Copy-on-Write of a File," U.S. Patent no. 7,085,909, August 2006.
11. Jason C. Young, Rajagopal Ananthanarayanan, Randal C. Burns, Darrell D. E. Long, Robert M. Rees, Ralph A. Becker-Szendy, James J. Seeger and David M. Wolf. "Managing File System Versions," U.S. Patent no. 7,139,781, November 2006.

Professional Activities

Invited Lectures

- 2019 "Geomancy: Automated Performance Enhancement through Data Placement Optimization," IBM Research, Zürich, Organisation Européenne pour la Recherche Nucléaire (CERN), Sorbonne Université.
"Twizzler: An OS for Next-generation Memory Hierarchies," IBM Research, Zürich, Organisation Européenne pour la Recherche Nucléaire (CERN), Sorbonne Université.
- 2016 "RESAR: Reliable Storage at Exabyte Scale," IBM Research, Zürich, Université Paris–Descartes.
"Pilot: A Framework that Understands How to Do Performance Benchmarks the Right Way," Organisation Européenne pour la Recherche Nucléaire (CERN).
- 2014 "Horus: Fine-Grained Security for Large-Scale Storage," University of Houston.
"SNQoS: Scalable Shared-Nothing QoS for High-Performance Storage Systems," San Diego State University, University of New Orleans.
"Science and Cyber Security," Lockheed-Martin "Skunk Works" (Advanced Development Programs).
- 2013 "Horus: Fine-Grained Security for Large-Scale Storage," Laboratoire d'Informatique de Paris 6 (LIP6), Université Pierre et Marie Curie, Conservatoire National des Arts et Métiers, Université de Technologie en Sciences des Organisations et de la Décision de Paris–Dauphine.
"RESAR Storage: a System for Two-Failure Tolerant, Self-Adjusting Million Disk Storage Cluster," Conservatoire National des Arts et Métiers, Université Paris–Descartes, Université Pierre et Marie Curie (Laboratoire d'Informatique de Paris 6).
- 2012 "RESAR Storage: a System for Two-Failure Tolerant, Self-Adjusting Million Disk Storage Cluster," University of Oregon.
- 2011 "Science of Cyber Security," Sandia National Laboratories.
- 2010 "Scalable, Parallel Deduplication for Large Scale Data Systems," Pacific Northwest National Laboratory.
"Extreme Binning: Scalable, Parallel Deduplication for Chunk-based File Backup," Université de Technologie en Sciences des Organisations et de la Décision de Paris–Dauphine.
- 2009 "Maximizing Efficiency by Trading Storage for Computation," IBM-Amrita Cloud Symposium, Amrita University, Coimbatore, India.
"High-Performance Petascale File Systems," Université de Technologie en Sciences des Organisations et de la Décision de Paris–Dauphine.
"High-Performance Petascale File Systems," Commissariat à l'Energie Atomique, Direction des Applications Militaires et Direction des Ames Nucléaires.
- 2007 "Ceph: A Scalable, High-Performance Distributed File System," University of Technology, Sydney, Australia.
"Ceph: A Scalable, High-Performance Distributed File System," University of California, San Diego.

- 2006 "Storage Management – Unchecked Challenges, Untapped Opportunities," Workshop on Distributed Data and Structures, Santa Clara University.
- 2005 "Storage Management – Unchecked Challenges, Untapped Opportunities," Storage Networking Summit, Bangalore, India.
 "Ceph: A Distributed Object-based Storage System," Indian Institute of Science, Bangalore, India.
 "Deep Store: Minimum Redundancy Archival Storage," Amrita University, Coimbatore, India.
- 2004 "Peta-scale Data Storage: Challenges and (a few) Solutions," Yahoo!
 "Deep Store," University of Arizona.
 "Deep Store," Vrije Universiteit.
- 2003 "Deep Store: Associative Archival Storage Architecture," Network Appliance.
- 2002 "Group-based Management of Distributed File Caches," Hewlett-Packard Laboratories.
 "Group-based Management of Distributed File Caches," Technische Universität München.
- 2000 "Perspective on Academic Storage Research," IBM Research.
 "Produce-Consumer Locking" and "Analytical Lease Modeling," San Francisco State University.
 "The Storage Tank System," Université de technologie en sciences des organisations et de la décision de Paris–Dauphine
- 1997 "Differential Storage Management" and "Efficient Video-on-Demand Services," Université de technologie en sciences des organisations et de la décision de Paris–Dauphine
 "Differential Storage Management," Queen Mary College, University of London.
 "A Dynamic Disk Spin-down Technique for Mobile Computing," University of California, Riverside.
 "Replication with Minimal Metadata," IBM Corporation.
- 1996 "Predicting File System Actions from Prior Events," University of California, San Diego.
- 1995 "Fault-Tolerance in Distributed Systems," WilTel Advanced Technology Group.
- 1994 "The REINAS System," University of California, Riverside.
- 1992 "A Study of the Reliability of Internet Sites," San Diego State University.
- 1991 "Swift – A Distributed Storage System for Multimedia," Matsushita Information Technology Laboratory.
 "A Study of the Reliability of Internet Sites," INRIA (Institut National de Recherche en Informatique et Automatique).
 "A Study of the Reliability of Internet Sites," University of California, San Diego.
 "Swift – A Distributed Storage System for Multimedia," Hewlett-Packard Laboratories.
 "Reliable Distributed Storage Systems," Sun Microsystems.
 "A Study of the Reliability of Internet Sites," Bay Area Systems Seminar.
- 1990 "Swift: A Storage Architecture for Very Large Objects," Hewlett-Packard Laboratories.
 "Estimating the Reliability of Regeneration-Based Replica Control Protocols," IBM Almaden Research and SRI International.
- 1989 "Availability and Reliability of Regeneration Protocols for Replicated Objects," Hewlett-Packard Laboratories.

- 1988 “Optimistic Protocols for Replicated Data Management,” Vanderbilt University and Stanford University.
- 1987 “The Management of Replication in a Distributed Computing System,” Georgia Institute of Technology.
 “Optimistic Algorithms for Replicated Data Management,” Oxford University.
 “The Management of Replication: An Overview of the Gemini Project,” United States Naval Research Laboratory.
- 1986 “On Improving the Availability of Replicated Files,” University of California, Berkeley.

Conference Organization

- 2024 General Chair: *Conference on Mass Storage Systems and Technologies* (MSST). Program Committee: *Unix Technical Conference* (ATC).
- 2023 Program Committee: *Fourth Joint International Conference on Deep Learning, Big Data and Blockchain* (DBB).
- 2022 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2021 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2020 Program Chair: *Conference on Mass Storage Systems and Technologies* (MSST).
 Program Committee: *Symposium on File and Storage Systems Technology* (FAST), *International Conference on Cloud Computing and Services Science* (CLOSER 2020).
 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2019 Program Committee: *IEEE International Conference on Machine Learning and Applications* (ICMLA 2019).
 Reviewer: *International Conference on Cloud and Big Data Computing* (CBDC 2019).
 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2018 Program Committee: *IEEE Cluster Conference* (Cluster 2018).
 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2017 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2016 Program Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
 Steering Committee: *Petascale Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2015 Program Committee: *Symposium on File and Storage Systems Technology* (FAST).
 Steering Committee: *Petascale Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).

- 2014 Program Committee: *International Conference on Advanced Communications and Computation* (INFOCOMP), *Conference on Mass Storage Systems and Technologies* (MSST).
 General Chair: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
 Steering Committee: *Petascade Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
 Reviewer: *International Symposium on Performance Analysis of Systems and Software* (ISPASS).
- 2013 Reviewer: *Electronics and Telecommunications Research Institute Journal* (ETRI), *IEEE Transactions on Neural Networks and Learning Systems*, *International Workshop on Combinatorial Algorithms*.
 Program Committee: *Annual International Systems and Storage Conference* (SYSTOR), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Conference on Mass Storage Systems and Technologies* (MSST).
 Steering Committee: *Petascade Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
 Advisory Committee: *International Conference on Electronics, Computing and Communication Technologies* (CONECCT).
- 2012 Steering Committee: *Petascade Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
 Program Committee: *Symposium on File and Storage Systems Technology* (FAST).
- 2011 Steering Committee: *Petascade Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
 Program Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2010 Program Chair: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
 Steering Committee: *Petascade Data Storage Workshop* (PDSW), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
- 2009 Program Committee: *International Workshop on Software Support for Portable Storage* (IWSSPS), *Inaugural International Conference on Virtualization and Cloud Computing*, *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Petascade Data Storage Workshop* (PDSW).
 Program Chair: *Web Information Systems Engineering* (WISE).
 General Chair: *Symposium on Applications and the Internet* (SAINT).
 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
- 2008 Program Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Petascade Data Storage Workshop* (PDSW).
 Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).

- 2007 Program Committee: *Symposium on File and Storage Systems Technology* (FAST), *Petascale Data Storage Workshop* (PDSW).
Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS), *Symposium on File and Storage Systems Technology* (FAST).
- 2006 Program Committee: *International Conference on Dependable Systems and Networks* (DSN), *Dependable Computing and Communications Symposium* (DCCS), *Symposium on Applications and the Internet* (SAINT).
Steering Committee: *Symposium on File and Storage Systems Technology* (FAST), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2005 Program Committee: *Symposium on File and Storage Systems Technology* (FAST), *Workshop on Storage Security and Survivability* (StorageSS), *First International Workshop on Software Support for Portable Storage* (IWSSPS).
- 2004 Program Committee: *Symposium on Applications and the Internet* (SAINT); SIGMetrics.
Steering Committee: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2003 Program Committee: *Usenix Technical Conference*, *Symposium on File and Storage Systems Technology* (FAST), *Very Large Data Bases* (VLDB), *International Conference on Distributed Computing Systems* (ICDCS), *IEEE International Performance, Computing, and Communications Conference* (IPCCC), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Steering Committee: *Symposium on File and Storage Systems Technology* (FAST), *Workshop on Mobile Computing Systems and Applications* (WMCSA), *IEEE International Performance, Computing, and Communications Conference* (IPCCC), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 2002 Program Committee: *IEEE International Performance, Computing, and Communications Conference* (IPCCC), *Workshop on Caching, Coherency and Consistency* (WC3), *Workshop on Distributed Data and Structures* (WDAS), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Steering Committee: *Symposium on File and Storage Systems Technology* (FAST), *Workshop on Mobile Computing Systems and Applications* (WMCSA), *IEEE International Performance, Computing, and Communications Conference* (IPCCC).
- 2001 Founding Program Chair: *Symposium on File and Storage Systems Technology* (FAST).
Program Chair: *IEEE International Performance, Computing, and Communications Conference* (IPCCC).
Program Committee: *International Conference on Distributed Computing Systems* (ICDCS).
- 2000 Program Committee: *Communication Networks and Distributed Systems Modeling and Simulation Conference* (CNDS), *Research Issues in Data Engineering* (RIDE), *Hot Interconnects*, *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Steering Committee: *Workshop on Mobile Computing Systems and Applications* (WMCSA), *Symposium on Applications and the Internet* (SAINT).
- 1999 General Chair: *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Program Committee: *Communication Networks and Distributed Systems Modeling and Simulation Conference* (CNDS), *IEEE International Performance, Computing, and Communications Conference* (IPCCC).

- Steering Committee: *Symposium on Applications and the Internet* (SAINT).
- 1998 Program Committee: *Symposium on Reliable Distributed Systems* (SRDS), *Fault Tolerant Computing Systems* (FTCS), *Usenix Technical Conference*.
Steering Committee: *IEEE Workshop on Mobile Computing Systems and Applications* (WMCSA).
Advisory Committee: *Hawaii International Conference on System Sciences* (HICSS).
- 1997 General Chair: *IEEE Computer Society CompCon*.
Program Committee: *International Conference on Parallel and Distributed Systems* (ICPDS), *Communications Networks and Distributed Systems Modeling and Simulation Conference* (CNDS), *Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Steering Committee: *Hot Topics in Operating Systems VI* (HotOS).
- 1996 General Chair: *IEEE Computer Society/National Science Foundation Workshop on Workspaces in the Information Age*.
- 1996 Program Chair: *IEEE Computer Society CompCon*.
Program Committee: *International Conference on Distributed Computing Systems* (ICDCS), *Annual Simulation Symposium*, *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
Steering Committee: *Hot Topics in Operating Systems V* (HotOS).
- 1995 Program Committee: *Symposium on Reliable Distributed Systems* (SRDS), *Usenix Technical Conference*, *Services in Distributed and Networked Environments* (SDNE), *International Conference on Distributed Computing Systems* (ICDCS), *Annual Simulation Symposium*, *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 1994 Founding General Chair: *Workshop on Mobile Computing Systems and Applications* (WMCSA).
- 1994 Program Committee: *Applied Data Bases*; *First International Workshop on Services in Distributed and Networked Environments*, *International Conference on Distributed Computing Systems* (ICDCS), *Annual Simulation Symposium*, *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 1993 Program Committee: *International Conference on Parallel and Distributed Systems* (ICPDS), *IEEE Workshop on Advances in Parallel and Distributed Systems*, *International Conference on Distributed Computing Systems* (ICDCS), *Annual Simulation Symposium*, *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 1992 Program Committee: *Annual Simulation Symposium*, *Second Workshop on the Management of Replicated Data* (WMRD), *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems* (MASCOTS).
- 1991 Local Arrangements Chair and Program Committee: *Second Workshop on the Management of Replicated Data* (WMRD).
- 1990 Publicity Chair: *First Workshop on the Management of Replicated Data* (WMRD).
- 1989 Program Committee: *Usenix Association/ACM/IEEE-CS*, *Workshop on Experiences with Building Distributed and Multiprocessor Systems*.
- 1988 Program Committee: *IFIP Working Group 6.5*, *International Working Conference on Message Handling Systems and Distributed Applications*.

Membership in Professional Associations

Year	Rank	Professional Society
2019	Member	International Association for Cryptologic Research (IACR)
2012	Member	Armed Forces Communications & Electronics Association
2008	Fellow	American Association for the Advancement of Science (AAAS)
2006	Fellow	Institute of Electrical & Electronics Engineers (IEEE)
2006	Member	American Society for Engineering Education (ASEE)
1991	Member	Usenix Association
1989	Member	Sigma Xi (Scientific Research Society)
1987	Member	IEEE Computer Society
1981	Member	Association for Computing Machinery (ACM)

Consulting

Years	Organization	Role
2015–	Wilson, Sonsini, Goodrich & Rosati	Technical Expert
	Fish & Richardson	Technical Expert
2015	Foley & Lardner	Technical Expert
	DLA Piper	Technical Expert
2014–	Latham & Watkins	Technical Expert
2014	Beck & Hall	Technical Expert
2013–15	Faegre, Baker & Daniels	Technical Expert
	Wilmer Cutler Pickering Hale	Technical Expert
2013–14	County of Tehama	Technical Expert
2012–13	Klarquist Sparkman	Technical Expert
	Wilson, Sonsini, Goodrich & Rosati	Technical Expert
2011–14	Morgan, Lewis & Bockius	Technical Expert
	Fish & Richardson	Technical Expert
2011–13	Connolly Bove	Technical Expert
2011–12	Haynes & Boone	Technical Expert
2011	Fujitsu Semiconductor America	Consultant
2010–12	Quinn Emanuel Urquhart & Sullivan	Technical Expert
	Fish & Richardson	Technical Expert
	Simpson, Thacher & Bartlett	Technical Expert
2009	Preston, Gates & Ellis	Technical Expert
2008–09	Weil, Gotshal & Mange	Technical Expert
2008–10	Hewlett-Packard Laboratories	Consultant
2007–	Center for Communications Research	Adjunct
2006–10	Weil, Gotshal & Manges	Technical Expert
2006–14	Fish & Richardson	Technical Expert
2004–05	Faegre & Benson	Technical Expert
2004	Siemens Technology-to-Business Center	Consultant
2002–	MITRE Corporation	JASON
2002–05	Fish & Richardson	Technical Expert
2001–03	Hecker Law Group	Technical Expert
	Bartlit, Beck, Herman, Palenchar & Scott	Technical Expert
	Jenner & Block	Technical Expert
2000	Sony Music Ventures	Consultant
1997–02	Chief of Naval Research	Consultant

1996	Chatham Group	Technical Expert
1995–11	IBM Almaden Research Center	Visiting Scientist

Editorial Duties

2023	Associate Editor, <i>Springer Journal of Big Data (JOBDB)</i>
2017–20	Editor-in-Chief, <i>Letters of the Computer Society</i> .
2016–19	Associate Editor, <i>IEEE Transactions on Computers</i> .
2011	Guest Editor, <i>Information Systems Databases: Creation, Management and Utilization</i> , Elsevier, vol. 36, no. 3, May 2011.
2010–16	Editor-in-Chief, <i>ACM Transactions on Storage</i> .
2009	Guest Editor, <i>Web Information Systems Engineering (WISE 2009)</i> , LNCS 5802.
2004–10	Associate Editor, <i>ACM Transactions on Storage</i> .
2002–03	Editorial Board, <i>Mobile Computing and Communications Review</i> .
1993–97	Editorial Board, <i>International Journal in Computer Simulation</i> .
1990–93	Editor, <i>Bulletin of the IEEE Computer Society Technical Committee on Operating Systems and Applications Environments</i> .
1990–93	Associate Editor, <i>International Journal in Computer Simulation</i> .

Reviewer of Technical Papers and Proposals

2023	ACM Transactions on Computer Systems.
2022	ACM Transactions on Computer Systems; The Journal of Super Computing; Performance Evaluation.
2021	Transactions on Dependable and Secure Computing.
2017	Information Security Journal: A Global Perspective; ACM Transactions on Storage.
2016	ACM Transactions on Modeling and Performance Evaluation of Computing Systems; Israel Science Foundation; Flanders Innovation & Entrepreneurship.
2012	ACM Transactions on Storage; Journal of Computer Science and Technology; IEEE Transactions on Neural Networks and Learning Systems; IEEE Transactions of Reliability.
2009	ACM Transactions on Storage; The Journal of Systems and Software.
2008	ACM Transactions on Storage; IEEE Transactions on Computers; International Conference on Networking, Architecture, and Storage.
2007	ACM Transactions on Storage; IEEE Transactions on Computers; Performance and Dependability Symposium (PDS).
2006	IEEE Transactions on Computers; ACM Transactions on Storage; IEEE Transactions on Magnetics; International Symposium on Stabilization, Safety, and Security of Distributed Systems; International Parallel and Distributed Processing Symposium; University of California, MICRO Program; International Conference on Dependable Systems and Networks.
2005	ACM Transactions on Storage; IEEE/ACM Transactions on Networking.

- 2004 ACM Transactions on Storage; IEEE Transactions on Computers; IEEE Transactions on Parallel and Distributed Systems; International Journal of Computers and Applications.
- 2003 Symposium on Operating Systems Principles; IEEE Transactions on Broadcasting; Computer Journal; IEEE Communications Letters.
- 2002 Transactions on Knowledge and Data Engineering; Transactions on Broadcasting; International Conference on Parallel Processing; IEEE Communications Letters; Symposium on Operating System Design and Implementation; Western Multiconference.
- 2001 IEEE Personal Communications; Workshop on Hot Topics in Operating Systems (HotOS); ETRI Journal; International Conference on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems; International Conference on Distributed Computing Systems; Symposium on Operating Systems Principles; Western Multiconference.
- 2000 Transactions on Knowledge and Data Engineering; Symposium on Reliable Distributed Systems; Communications of the ACM; IEEE Personal Communications; Transactions on Computer Systems; Transactions on Modeling and Computer Simulation; Workshop on Wireless Networks and Mobile Computing.
- 1999 ACM/Baltzer Mobile Networks and Applications; IEEE Software; IEEE Transactions on Software Engineering; International Conference on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems; Usenix Technical Conference; Journal of Parallel and Distributed Computing.
- 1998 IEEE Software; IEEE Transactions on Parallel and Distributed Systems; ACM Transactions on Computer Systems; Conference on Computing and Telecommunications; International Conference on High Performance Computing; International Conference on Parallel Processing.
- 1997 National Science Foundation; IEEE Computer; IEEE Transactions on Parallel and Distributed Systems; Symposium on Reliable Distributed Systems; Software – Practice & Experience; IEEE Software; Computer Journal; Transactions on Knowledge and Data Engineering.
- 1996 National Science Foundation; Computer Journal; University of California, MICRO program; Information Systems Journal; IEEE Transactions on Parallel and Distributed Systems.
- 1995 University of California, MICRO program; National Science Foundation; Computer Journal; Distributed Systems Engineering Journal; Computing Surveys; Dependable Computing and Critical Applications Conference; IEEE Transactions on Knowledge and Data Engineering.
- 1994 IEEE Computer; High Performance Distributed Computing; California Space Institute; Operating Systems Design and Implementation; International Journal in Computer Simulation; Information Processing Letters; Journal of Distributed and Parallel Data Bases; IEEE Software. ACM Transactions on Computer Systems.
- 1993 International Conference on Distributed Computing Systems; Distributed and Parallel Databases; Hawaii International Conference on Systems Sciences; International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems; International Symposium on Autonomous Decentralized Systems; IEEE Transactions on Parallel Distributed Systems; IEEE Transactions on Computers; Information Processing Letters; IEEE Computer.
- 1992 International Symposium on Autonomous Decentralized Systems; IEEE Transactions on Parallel Distributed Systems; IEEE/ACM Transactions on Networking; Information Systems; IFIP Dependable Computing in Critical Applications; IEEE Transactions on Software Engineering; IEEE Transactions on Computers; International Conference on Data Engineering; IEEE Transactions on Knowledge and Data Engineering; University of California, MICRO Program; IEEE Computer; Information Processing Letters.

- 1991 International Conference on Data Engineering; International Journal on Very Large Databases; ACM Conference on Principles of Data Base Systems; IEEE Transactions on Parallel and Distributed Systems; IEEE Transactions on Knowledge and Data Engineering; Computing Systems; University of California, MICRO Program; IEEE Computer; ACM Transactions on Computer Systems; Information Processing Letters.
- 1990 IEEE Transactions on Parallel and Distributed Systems; IEEE Transactions on Knowledge and Data Engineering; Computing Systems; Communications of the ACM; 1991 SIGMETRICS Conference on Measurement and Modeling of Computer Systems; University of California, MICRO Program; National Science Foundation; IEEE Computer; ACM Transactions on Computer Systems; Information Processing Letters; IEEE Software.
- 1989 National Science Foundation; IEEE Computer; ACM Transactions on Computer Systems; Information Processing Letters; IEEE Software; International Conference on Data Engineering; International Conference on Distributed Computing Systems.
- 1988 International Conference on Distributed Computing Systems.
- 1987 International Conference on Data Engineering.

Reviewer for Publishers

- 2011 John Wiley & Sons.
- 2003 Prentice-Hall.
- 2002 John Wiley & Sons.
- 1997 Academic Press; Addison-Wesley; Gordon and Beach.
- 1995 Addison-Wesley; Benjamin/Cummings; Springer-Verlag.
- 1994 Benjamin/Cummings; Morgan-Kaufman.
- 1993 Kluwer Academic Publishers; Norton; Benjamin/Cummings; McGraw-Hill; John Wiley & Sons.
- 1991–93 Aksen Associates.
- 1988–91 Addison-Wesley.

Government Service

- 2020 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Nuclear Security Administration. Department of Defense.
- 2019 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Nuclear Security Administration. Department of Defense. Department of State.
- 2018 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Nuclear Security Administration. Department of Defense.
- 2017 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Nuclear Security Administration. Department of Defense.

- 2016 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Security Directorate Advisory Committee (Pacific Northwest National Laboratory). Computer Information Science External Review Board (Sandia National Laboratory). National Nuclear Security Administration. Department of Energy. Department of Defense.
- 2015 Intelligence Science and Technology Experts Group (ISTEG) (National Academies of Sciences, Engineering and Medicine committee). National Security Directorate Advisory Committee (Pacific Northwest National Laboratory). Computer Information Science External Review Board (Sandia National Laboratory). National Nuclear Security Administration. Department of Energy. Department of Defense.
- 2014 National Security Directorate Advisory Committee (Pacific Northwest National Laboratory). Computer Information Science External Review Board (Sandia National Laboratory). National Nuclear Security Administration. Department of Energy. Department of Defense.
- 2013 National Research Council, Committee on Science and Technology for Defense Warning. National Nuclear Security Administration. Department of Energy. Department of Defense.
- 2012 National Research Council, Committee for Science and Technology Challenges to U.S. National Security Interests. National Research Council, Committee on Science and Technology for Defense Warning, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). National Nuclear Security Administration. Department of Energy. Department of Defense.
- 2011 National Research Council, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). National Research Council, report reviewer. National Nuclear Security Administration. Department of Defense. Department of the Navy.
- 2010 Department of Energy. National Nuclear Security Administration. Department of Defense. National Research Council, report reviewer.
- 2009 United States Army Laboratory Assessment Group. Department of Energy. National Nuclear Security Administration. Department of Defense. Department of Energy, Office of Science (Early Career panel).
- 2008 United States Army Laboratory Assessment Group. National Research Council, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). Department of Energy. National Nuclear Security Administration. Department of Defense.
- 2007 External Review Committee, United States Naval Postgraduate School. United States Army Technology Objectives Review Panel. National Science Foundation (review panels). National Research Council, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). National Research Council, Reviewer. Department of Energy. National Nuclear Security Administration. Department of Defense.
- 2006 National Science Foundation (review panels). National Research Council, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). National Research Council, Reviewer. Department of Energy. National Nuclear Security Administration. Department of Defense.
- 2005 National Research Council, Standing Committee on Technology Insight-Gauge, Evaluate, and Review (TIGER). Department of Energy. National Nuclear Security Administration. Department of Defense.
- 2004 National Research Council, Defense Intelligence Agency Technology Forecast and Reviews Committee. National Nuclear Security Administration. Department of Defense. Los Alamos National Laboratory, hiring committee for Associate Director for Nuclear Weapons Physics and Principal Associate Director for Nuclear Weapons Programs.

2003	National Science Foundation (two Information Technology Research panels). Department of Energy. National Nuclear Security Administration. Department of Defense.
2002	National Science Foundation (two Information Technology Research panels). Department of Energy. National Nuclear Security Administration.
2001	National Science Foundation. Department of Energy. National Nuclear Security Administration.
2000	National Science Foundation (Information Technology Research panel, National Environmental Observatory Network planning panel, CAREER panel).
1999	National Science Foundation (site review of Data Storage Systems Center, panel on Engineering Research Centers). Department of Defense.
1998	National Science Foundation (site review of Data Storage Systems Center). Department of the Navy.
1997	Department of the Navy.

Service to Professional Societies

2020	IEEE Fellows Evaluation Committee.
2014	IEEE Fellows Evaluation Committee.
2013	IEEE Fellows Evaluation Committee.
2011–12	Member and Chair Emeritus, IEEE Reynold B. Johnson Information Storage Systems Award Committee.
2011	IEEE Fellows Evaluation Committee, IEEE Fellow Society/Technical Council Evaluator.
2010	IEEE Fellows Evaluation Committee.
2008–11	Chair, IEEE Reynold B. Johnson Information Storage Systems Award Committee, IEEE Technical Field Awards Council.
2007	IEEE Fellows Evaluation Committee.
2006–08	IEEE Reynold B. Johnson Information Storage Systems Award Committee.
2000–02	Chair, Usenix Association Scholarship and Research Committee.
1998–02	Usenix Association Scholarship and Research Committee.
1997–	University Liaison for the Usenix Association.
1997	Committee on the Future of the Usenix Association.
1996–	Executive Committee, <i>IEEE Technical Committee on Operating Systems</i> .
1995	IEEE/ACM focus group on Software Engineering Body of Knowledge and Recommended Practices.
1993–96	Chair, <i>IEEE Technical Committee on Operating Systems and Applications Environments</i> .

University Service

Senate Service

- 2018–19 Committee on Research.
- 2016–17 Committee on Academic Freedom.
- 2008–09 Committee on Academic Personnel *ad hoc* subcommittee.
- 2007–08 Committee on Academic Personnel *ad hoc* subcommittee.
- 2006–07 Committee on Research.
- 2005–06 Committee on Research; Committee on Academic Personnel *ad hoc* subcommittee.
- 2003–04 Committee on Committees; Committee on Academic Personnel *ad hoc* subcommittee.
- 2001–02 Committee on Academic Personnel *ad hoc* subcommittee; Committee on Research.
- 2000–01 Committee on Research; Chair, Committee on Academic Personnel *ad hoc* subcommittee;
- 2000–01 Silicon Valley Regional Center Task Force; California Institutes for Science and Innovation proposal (with P. Mantey & Berkeley colleagues).
- 1999–00 Committee on Academic Personnel *ad hoc* subcommittee; Computing and Telecommunications Committee.
- 1998–99 Computing and Telecommunications Committee.
- 1995–97 Senate Rules Committee, Information Technology Disaster Preparedness Task Force.

Campus Service

- 2015– Genomics Institute Steering Committee.
- 2004–05 Confidential personnel review for senior administrator.
- 2000–04 Chair, Engineering II Building Committee.
- 1998–99 Confidential personnel review for senior administrator.
- 1998 Faculty mentor, California Alliance for Minority Participation (CAMP).
- 1997 Faculty mentor, California Alliance for Minority Participation (CAMP).
- 1996–97 Information Technology Disaster Preparedness Task Force.
- 1994–95 Task Force on Computer and Network Security, Summer Science Honors Institute.
- 1990–91 Technical Consultant to Computing and Telecommunication Services (CATS) for University computing facilities.
- 1988–89 Applied Sciences Building Remodeling Committee. Open Access Computing Committee.

School of Engineering Service

- 2017– Strategic Faculty Advisor to the School of Engineering.
- 2004–10 Associate Dean for Research and Graduate Studies.
- 1998–01 Associate Dean for Research.
- 1995–96 Electrical Engineering/Applied and Engineering Mathematics Committee.
- 1988–90 Natural Sciences Division Space Committee.

Departmental Service

- 2015–16 Personnel Review Committee; Chair, *ad hoc* Personnel Committee; Security & Privacy faculty hiring committee.
- 2014–15 Personnel Review Committee; Chair, *ad hoc* Personnel Committee; Experimental Computer Science faculty hiring committee.
- 2013–14 Department Chair Selection Committee; Personnel Review Committee; Data Sciences faculty hiring committee.
- 2011–12 Personnel Best Practices Committee; Personnel Review Committee; Chair, *ad hoc* Personnel Committee.
- 2010–11 Department Awards Coordinator; Chair, *ad hoc* Personnel Committee; Chair, Adjunct Committee.
- 2009–10 Director for Research, Computer Science Department; Chair, Adjunct Committee.
- 2008–09 Director for Research, Computer Science Department; Chair, Adjunct Committee.
- 2007–08 Director for Research, Computer Science Department; Chair, Adjunct Committee; Seminar Coordinator, Computer Science Department.
- 2006–07 Personnel Review Committee; Director for Research, Computer Science Department.
- 2005–06 Personnel Review Committee; Director for Research, Computer Science Department; Department of Information Systems and Technology Management.
- 2004–05 Director for Research, Computer Science Department; Master of Ceremonies, Engineering 2 dedication; Information Technologies Institute (ITI) Executive Committee.
- 2003–04 Director for Research, Computer Science Department; Information Technologies Institute (ITI) Executive Committee; School of Engineering Technical Services Committee.
- 2002–03 Chair, Promotion Review Subcommittee; Computer Science Faculty Recruitment Committee; ISTM Chair Recruitment; Faculty Recruitment Committee.
- 2001–02 Chair, Promotion Review Subcommittee; Computer Science Faculty Recruitment Committee; Computer Science Department Facilities Director; Computing Infrastructure Committee; Faculty Recruitment Committee.
- 2000–01 Computer Science Faculty Recruitment Committee; Computer Science Department Facilities Director; Computing Infrastructure Committee; Faculty Recruitment Committee.
- 1999–00 Computer Science Faculty Recruitment Committee; Computer Science Department Facilities Director; Computing Infrastructure Committee; Faculty Recruitment Committee.

- 1998–99 Computer Science Faculty Recruitment Committee; Computer Science Department Facilities Director; Computing Infrastructure Committee; Faculty Recruitment (Software Engineering); Information Systems Management Major.
- 1997–98 Computer Science Faculty Recruitment Committee; Computer Science Department Facilities Director; Information Systems Management Major.
- 1991–96 Computer Science Facilities Director.
- 1990–91 Computer & Information Sciences Assistant Facilities Director.
- 1989–90 Computer & Information Sciences Department Faculty Recruitment Committee, Co-Chair for the Operating Systems position.
- 1989 Computer & Information Sciences Department Representative to Electronics Engineering Program Development Committee.
- 1988–89 Research Funding Director, Computer & Information Sciences Department (responsible for the CIS/CE 1989 grant proposal to the National Science Foundation Institutional Infrastructure Program).
- 1988–89 Computer & Information Sciences Department Representative on Computer Engineering Department Recruitment Committee.

System-wide Service

- 2020–23 Academic Council's Special Committee on Laboratory Issues (ACSCOLI).
- 2011–16 Academic Council's Special Committee on Laboratory Issues (ACSCOLI).
- 2007–13 Science & Technology Committee for both Lawrence Livermore and Los Alamos National Laboratories (University of California, Office of the President).
- 2007–14 Multiple Directorate and Capability review committees for Lawrence Livermore and Los Alamos National Laboratories (University of California, Office of the President).
- 2007–08 Advisory Board, Department of Computer Engineering, Santa Clara University; Predictive Knowledge Systems Advisory Committee, Lawrence Livermore National Laboratory (University of California, Office of the President).
- 2007 Nonproliferation, Homeland and International Security, Directorate Review Committee, Lawrence Livermore National Laboratory; Petascale Computing Management Review Committee, Los Alamos National Laboratory (University of California, Office of the President).
- 2005–20 Faculty Advisory Group for the South Asia Initiative.
- 2005– Executive Committee, Center for Information technology Research in the Interest of Society (CITRIS).
- 2003–07 National Security Panel and Intelligence Panel for the National Laboratories (University of California, Office of the President).
- 2002–03 Chair, University-wide Committee on Research Policy; National Security Panel, Science & Technology Panel, and President's Council (University-wide Senate Representative).
- 2001–02 Vice Chair, University-wide Committee on Research Policy; Screening Committee for the Director of Lawrence Livermore National Laboratory; National Security Panel, Science & Technology Panel, President's Council (University-wide Senate Representative).
- 2000–03 University-wide Committee on Research Policy.

TEACHING 1988–89

				Enrolled	%Eval Retd	Shared?
Fall	CIS	104	Introduction to Compiler Design	43	74	no
	CIS	301	Supervised Teaching Experience	1	n/a	no
Winter	CIS	221	Advanced Operating Systems	5	100	no
Spring	CIS	104	Introduction to Compiler Design	31	74	no
	CIS	301	Supervised Teaching Experience	1	100	no

TEACHING 1989–90

				Enrolled	%Eval Retd	Shared?
Fall	CIS	204	Compiler Design	18	94	no
	CIS	297	Independent Study/Research	1	n/a	no
	CIS	299	Thesis Research	1	n/a	no
Winter	CIS	104	Introduction to Compiler Design	17	88	no
	CIS	195	Independent Study/Research	1	n/a	no
	CIS	221	Advanced Operating Systems	8	88	no
	CIS	299	Thesis Research	3	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no
Spring	CIS	104	Introduction to Compiler Design	22	86	no
	CIS	297	Independent Study/Research	2	n/a	no
	CIS	299	Thesis Research	3	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no

TEACHING 1990–91

				Enrolled	%Eval Retd	Shared?
Fall	CIS	204	Compiler Design	10	100	no
	CIS	221	Advanced Operating Systems	12	75	no
	CIS	299	Thesis Research	3	n/a	no
Winter	CIS	104A	Fundamentals of Compiler Design I	18	67	no
	CIS	198	Independent Study/Research	2	n/a	no
	CIS	297	Independent Study/Research	1	n/a	no
	CIS	299	Thesis Research	3	n/a	no
	CIS	301	Supervised Teaching Experience	1	n/a	no
Spring	CIS	104B	Fundamentals of Compiler Design II	17	76	no
	CIS	195	Independent Study/Research	2	n/a	no
	CIS	297	Independent Study/Research	2	n/a	no

TEACHING 1991–92

				Enrolled	%Eval Retd	Shared?
Fall	CIS	221	Advanced Operating Systems	9	100	no
	CIS	198	Independent Study/Research	1	n/a	no
	CIS	297	Independent Study/Research	2	n/a	no
	CIS	299	Thesis Research	1	n/a	no
Winter	CIS	198	Independent Study/Research	2	n/a	no
	CIS	297	Independent Study/Research	3	n/a	no

Spring	CIS	299	Thesis Research	3	n/a	no
	CIS	104B	Fundamentals of Compiler Design II	14	79	no
	CIS	195	Independent Study/Research	2	n/a	no
	CIS	198	Independent Study/Research	1	n/a	no
	CIS	203	Programming Languages & Environments	8	88	no
	CIS	297	Independent Study/Research	3	n/a	no
	CIS	299	Thesis Research	5	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no

TEACHING 1992–93

				Enrolled	%Eval Retd	Shared?
Fall	CIS	111	Introduction to Operating Systems	43	80	no
	CIS	299	Thesis Research	6	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no
Winter	CIS	198	Independent Study/Research	1	n/a	no
	CIS	204	Compiler Design	5	100	no
	CIS	297	Independent Study/Research	4	n/a	no
	CIS	299	Thesis Research	3	n/a	no
Spring	CIS	195	Senior Thesis Research	1	n/a	no
	CIS	221	Advanced Operating Systems	8	100	no
	CIS	297	Independent Study/Research	4	n/a	no
	CIS	299	Thesis Research	3	n/a	no
	Stev	003	Privacy & Computer Security	11	73	no

TEACHING 1993–94

				Enrolled	%Eval Retd	Shared?
Fall	CIS	111	Introduction to Operating Systems	36	78	no
	CIS	297	Independent Study/Research	4	n/a	no
	CIS	299	Thesis Research	3	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no
Winter	CIS	204	Compiler Design	5+1	100	no
	CIS	297	Independent Study/Research	10	n/a	no
	CIS	299	Thesis Research	1	n/a	no
Spring	CIS	203	Programming Languages & Environments	21	85	no
	CIS	297	Independent Study/Research	8	n/a	no
	CIS	299	Thesis Research	5	n/a	no

TEACHING 1994–95

				Enrolled	%Eval Retd	Shared?
Fall	CIS	111	Introduction to Operating Systems	56	68	no
	CIS	195	Senior Thesis Research	1	n/a	no
	CIS	221	Advanced Operating Systems	15	60	no
	CIS	297	Independent Study/Research	3	n/a	no
	CIS	299	Thesis Research	9	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no
Winter	CIS	198	Independent Study/Research	3	n/a	no

Spring	CIS	297	Independent Study/Research	5	n/a	no
	CIS	299	Thesis Research	10	n/a	no
	CIS	195	Senior Thesis Research	1	n/a	no
	CIS	203	Programming Languages & Environments	16+3	95	no
	CIS	297	Independent Study/Research	2	n/a	no
	CIS	299	Thesis Research	10	n/a	no
	CIS	301	Supervised Teaching Experience	1	100	no

TEACHING 1995–96

				Enrolled	%Eval Retd	Shared?
Fall	CIS	111	Introduction to Operating Systems	18	78	no
	CIS	204	Compiler Design	6	83	no
	CIS	297	Independent Study/Research	3	n/a	no
	CMPE	297	Independent Study/Research	1	n/a	no
Winter	CIS	299	Thesis Research	3	n/a	no
	CIS	198	Independent Study/Research	2	n/a	no
	CIS	297	Independent Study/Research	4	n/a	no
	CIS	299	Thesis Research	3	n/a	no
Spring	CIS	195	Senior Thesis Research	1	n/a	no
	CIS	221	Advanced Operating Systems	13	77	no
	CIS	297	Independent Study/Research	3	n/a	no
	CMPE	297	Independent Study/Research	1	n/a	no
	CIS	299	Thesis Research	6	n/a	no
	CMPE	299	Thesis Research	1	n/a	no

TEACHING 1996–97

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	111	Introduction to Operating Systems	20	70	no
	CMPS	195	Thesis Research	1	n/a	no
	CMPS	297	Independent Study/Research	8	n/a	no
	CMPE	297	Independent Study/Research	4	n/a	no
	CMPS	299	Thesis Research	5	n/a	no
	CMPE	299	Thesis Research	3	n/a	no
	CMPS	301	Supervised Teaching Experience	1	100	no
Winter	CMPS	290X	Cryptography	6	67	no
	CMPS	297	Independent Study/Research	5	n/a	no
	CMPS	299	Thesis Research	6	n/a	no
	CMPE	299	Thesis Research	2	n/a	no
Spring	CMPS	090X	Internet Security & Privacy	19	58	no
	CMPS	221	Advanced Operating Systems	8	88	no
	CMPS	297	Independent Study/Research	6	n/a	no
	CMPS	299	Thesis Research	6	n/a	no
	CMPE	299	Thesis Research	3	n/a	no

TEACHING 1997–98

Enrolled %Eval Retd Shared?

Fall	CMPS		Sabbatical Leave			
	CMPS	297	Independent Study/Research	1	n/a	no
	CMPS	299	Thesis Research	9	n/a	no
Winter	CMPE	299	Thesis Research	3	n/a	no
	CMPS	112	Comparative Programming Languages	58	60	no
	CMPS	297	Independent Study/Research	3	n/a	no
	CMPS	299	Thesis Research	10	n/a	no
	CMPE	299	Thesis Research	2	n/a	no
Spring	CMPS	301	Supervised Teaching Experience	1	n/a	no
	CMPS	221	Advanced Operating Systems	9	100	no
	CMPS	297	Independent Study/Research	2	n/a	no
	CMPS	299	Thesis Research	8	n/a	no
	CMPE	299	Thesis Research	1	n/a	no

TEACHING 1998-99

				Enrolled	%Eval Retd	Shared?
Fall	CMPE	299	Thesis Research	1	n/a	no
	CMPS	297	Independent Study/Research	3	n/a	no
	CMPS	299	Thesis Research	6	n/a	no
Winter	CMPE	299	Thesis Research	1	n/a	no
	CMPS	297	Independent Study/Research	4	n/a	no
	CMPS	299A	Thesis Research	6	n/a	no
Spring	CMPE	299	Thesis Research	1	n/a	no
	CMPS	111	Introduction to Operating Systems	62	76	no
	CMPS	221	Advanced Operating Systems	6	100	no
	CMPS	297	Independent Study/Research	2	n/a	no
	CMPS	299	Thesis Research	5	n/a	no
Summer	CMPS	301	Supervised Teaching Experience	1	100	no
	CMPS	198	Independent Study/Research	1	n/a	no

TEACHING 1999-00

				Enrolled	%Eval Retd	Shared?
Fall	CMPE	299	Thesis Research	1	n/a	no
	CMPS	111	Introduction to Operating Systems	45	63	no
	CMPS	297	Independent Study/Research	2	n/a	no
	CMPS	299	Thesis Research	6	n/a	no
	CMPS	301	Supervised Teaching Experience	1	100	no
Winter	CMPS	112	Comparative Programming Languages	47	73	no
	CMPS	290X	Cryptography	3	0	no
	CMPS	297	Independent Study/Research	3	n/a	no
	CMPS	299	Thesis Research	2	n/a	no
Spring	CMPS	198	Independent Study/Research	1	n/a	no
	CMPS	297	Independent Study/Research	2	n/a	no
	CMPS	299	Thesis Research	3	n/a	no

TEACHING 2000-01

Enrolled %Eval Retd Shared?

Fall	CMPS	299C	Thesis Research	2	n/a	no
	CMPS	301	Supervised Teaching Experience	1	n/a	no
	ISM	150	Introduction To Information Networks	36	0	no
Winter	CMPS	112	Comparative Programming Languages	75	77	no
	CMPS	297A	Individual Study/ Research	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
	CMPS	301	Supervised Teaching Experience	1	n/a	no
Spring	CMPS	297A	Individual Study/ Research	3	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
Summer	CMPS	297A	Individual Study/ Research	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no

TEACHING 2001-02

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	198	Individual Study/Research	2	n/a	no
	CMPS	297A	Individual Study/Research	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
	ISM	150	Introduction To Information Networks	54	69	no
Winter	CMPS	195	Senior Thesis Research	1	n/a	no
	CMPS	198	Individual Study/Research	1	n/a	no
	CMPS	221	Advanced Operating Systems	17	82	no
	CMPS	221-50	Advanced Operating Systems	9	89	no
	CMPS	297A	Individual Study/Research	2	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
	CMPS	112	Comparative Programming Languages	62	61	no
Spring	CMPS	198	Individual Study/Research	1	n/a	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study/Research	1	n/a	no
	CMPS	297B	Individual Study/Research	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
	CMPS	297A	Individual Study/Research	1	n/a	no
Summer	CMPS	297A	Individual Study/Research	1	n/a	no

TEACHING 2002-03

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	111	Introduction To Operating Systems	53	92	no
	CMPS	280S	Seminar On Computer Systems	10	n/a	no
	CMPS	297A	Individual Study/Research	2	n/a	no
	CMPS	297B	Individual Study/Research	1	n/a	no
Winter	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study/Research	3	n/a	no
	CMPS	297B	Individual Study/Research	1	n/a	no
Spring	CMPE	299B	Thesis Research	1	n/a	no
	CMPS	229	Storage Systems	8	100	no
	CMPS	299B	Thesis Research	1	n/a	no

TEACHING 2003–04

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	13H	Introduction Programming/Data Structures	9	77	no
	CMPS	280S	Seminar Computer Systems	14	0	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
Winter	CMPS	111	Introduction Operating Systems	47	70	no
	CMPS	297A	Individual Study	3	n/a	no
	CMPS	299B	Thesis Research	2	n/a	no
Spring	CMPE	297B	Individual Study	1	n/a	no
	CMPS	290S	Adv Topics: Computer Systems	7	100	no
	CMPS	297A	Individual Study	2	n/a	no
	CMPS	299A	Thesis Research	2	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no

TEACHING 2004–05

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
Winter	CMPE	299A	Thesis Research	1	n/a	no
	CMPE	299B	Thesis Research	1	n/a	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	2	n/a	no
Spring	CMPS	280S	Seminar Computer Systems	18	50	no
	CMPS	111	Introduction to Operating Systems	58	76	no
	CMPS	297A	Individual Study	2	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
Summer	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no

TEACHING 2005–06

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	2	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299B	Thesis Research	2	n/a	no
	CMPS	299C	Thesis Research	1	n/a	no
Winter	CMPS	60N	Beginning Programming	3	133	no
	CMPS	299A	Thesis Research	2	n/a	no
	CMPS	299B	Thesis Research	3	n/a	no
	CMPS	299C	Thesis Research	1	n/a	no
Spring	CMPS	111	Introduction to Operating Systems	24	108	yes

CMPS	280	Seminar on Computer Systems	13	8	no
CMPS	299A	Thesis Research	3	n/a	no
CMPS	299B	Thesis Research	2	n/a	no
CMPS	299C	Thesis Research	1	n/a	no

TEACHING 2006-07

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	299B	Thesis Research	2	n/a	no
	CMPS	299B	Thesis Research	2	n/a	no
Winter	CMPS	299A	Thesis Research	2	n/a	no
	CMPS	299B	Thesis Research	2	n/a	no
Spring	CMPS	111	Introduction to Operating Systems	14	86	no
	CMPS	198	Individual Study or Research	1	n/a	no
	CMPS	280S	Seminar on Computer Systems	16	13	no
	CMPS	299B	Thesis Research	2	n/a	no

TEACHING 2007-08

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	221	Advanced Operating Systems	10	90	no
	CMPS	280S	Seminar on Computer Systems	10	50	no
	CMPS	299B	Thesis Research	2	n/a	no
Winter	CMPS	297A	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	1	n/a	no
Spring	CMPS	111	Introduction to Operating Systems	38	71	no
	CMPS	198	Individual Study or Research	1	n/a	no
	CMPS	280S	Seminar on Computer Systems	13	62	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	1	n/a	no

TEACHING 2008-09

				Enrolled	%Eval Retd	Shared?
Fall	CMPE	297B	Individual Study	1	n/a	no
	CMPS	111	Introduction to Operating Systems	21	43	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
Winter	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
Spring	CMPS	223	Advanced Computer Security	12	83	no
	CMPS	223	Advanced Computer Security (LANL)	2	100	no
	CMPS	297A	Individual Study	2	n/a	no

CMPS	297B	Individual Study	1	n/a	no
CMPS	299C	Thesis Research	1	n/a	no

TEACHING 2009–10

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	5P	Introduction to Programming in Python	92	48	no
	CMPS	280S	Seminar on Computer Systems	16	75	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	297C	Individual Study	2	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
Winter	CMPS	297A	Individual Study	2	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no
Spring	CMPS	198F	Independent Study or Research	1	n/a	no
	CMPS	229	Storage Systems	11	82	no
	CMPS	280S	Seminar on Computer Systems	14	36	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	3	n/a	no

TEACHING 2010–11

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	5C	Introduction to Programming in C/C++	119	45	no
	CMPS	280S	Seminar on Computer Systems	12	33	yes
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299C	Thesis Research	3	n/a	no
Winter	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
Spring	CMPS	111	Introduction to Operating Systems	28	64	no
	CMPS	280S	Seminar on Computer Systems	13	69	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	3	n/a	no

TEACHING 2011–12

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	198	Independent Study or Research	1	n/a	no
	CMPS	280S	Seminar on Computer Systems	20	40	yes
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
Winter	CMPS	229	Storage Systems	13	38	no

Spring	CMPS	297A	Individual Study	2	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
	CMPS	299C	Thesis Research	1	n/a	no
	CMPS	5P	Introduction to Programming in Python	136	66	no
	CMPS	280S	Seminar on Computer Systems	19	47	no
	CMPS	296	Masters Project	1	n/a	no
Summer	CMPS	297A	Individual Study	2	n/a	no
	CMPS	297B	Individual Study	2	n/a	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
	CMPS	297A	Individual Study	3	n/a	no
	CMPS	299	Thesis Research	1	n/a	no

TEACHING 2012–13

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	280S	Seminar on Computer Systems	17	76	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	2	n/a	no
	CMPS	299C	Thesis Research	6	n/a	no
Winter	CMPS	297A	Individual Study	2	n/a	no
	CMPS	297B	Individual Study	3	n/a	no
	CMPS	297C	Individual Study	2	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
Spring	CMPS	299C	Thesis Research	5	n/a	no
	CMPS	280S	Seminar on Computer Systems	14	71	no
	CMPS	290X	Cryptograph/Security	7	43	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
	CMPS	297B	Individual Study	3	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	6	n/a	no

TEACHING 2013–14

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	105	Systems Programming	18	22	no
	CMPS	280S	Seminar on Computer Systems	13	23	yes
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
Summer	CMPS	297	Independent Study or Research	1	n/a	no
	CMPS	297A	Individual Study	1	n/a	no
Winter	CMPS	111	Introduction to Operating Systems	32	59	no
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	2	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	4	n/a	no

Spring	EE	195	Senior Thesis Research	1	n/a	no
	CMPS	280S	Seminar Computer Systems	14	50	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	2	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no

TEACHING 2014–15

				Enrolled	%Eval Retd	Shared?
Fall	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
Winter	CMPS	111	Introduction to Operating Systems	42	69	no
	CMPS	280S	Seminar Computer Systems	11	45	yes
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	5	n/a	no
Spring	CMPS	223	Advanced Computer Security	7	86	no
	CMPS	195	Senior Thesis Research	1	n/a	no
	CMPS	280S	Seminar Computer Systems	11	45	yes
	CMPS	297B	Individual Study	1	n/a	no
	CMPS	299B	Thesis Research	1	n/a	no
	CMPS	299C	Thesis Research	6	n/a	no

TEACHING 2015–16

				Enrolled	%Eval Retd	Shared?
Summer	CMPS	299A	Thesis Research	1	n/a	no
Fall	CMPS	13H	Honors Introduction to Programming	40	68	no
	CMPS	13L	Honors Introduction to Programming Laboratory	17	n/a	no
	CMPS	13L	Honors Introduction to Programming Laboratory	19	n/a	no
	CMPS	280S	Seminar Computer Systems	9	44	yes
	CMPS	299C	Thesis Research	4	n/a	no
Winter	DANM	297	Independent Study	1	n/a	no
	CMPS	111	Introduction to Operating Systems	77	34	no
	CMPS	280S	Seminar Computer Systems	7	29	yes
	CMPS	299C	Thesis Research	3	n/a	no
Spring	CMPS	198F	Independent Study or Research	1	n/a	no
	CMPS	229	Storage Systems	9	89	no
	CMPS	280S	Seminar Computer Systems	14	64	yes
	CMPS	299C	Thesis Research	4	n/a	no

TEACHING 2016–17

				Enrolled	%Eval Retd	Shared?
Summer	CMPE	198	Independent Study or Research	1	n/a	no
	CMPS	193	Field Study	1	n/a	no
Fall	CMPS	296	Masters Project	1	n/a	no
	CMPS	299C	Thesis Research	2	n/a	no

Winter	CMPS	111	Introduction to Operating Systems	115	37	no
	CMPS	280S	Seminar Computer Systems	10	60	no
	CMPS	299C	Thesis Research	2	n/a	no
Spring	CMPE	297A	Individual Study	1	n/a	no
	CMPS	12B	Introduction to Data Structures	339	27	no
	CMPS	12M	Data Structures Laboratory	317	28	no
	CMPS	280S	Seminar Computer Systems	11	36	no

TEACHING 2017–18

				Enrolled	%Eval Retd	Shared?
Summer	CMPE	198	Individual Study or Research	1	n/a	no
Fall			Medical Leave			
Winter	CMPE	194F	Dangerous World	14	n/a	no
	CMPE	198F	Individual Study or Research	3	n/a	no
	CMPS	280S	Seminar Computer Systems	7	43	no
	CMPS	111	Introduction to Operating Systems	34	26	no
	CMPS	296	Masters Project	1	n/a	no
	CMPS	297A	Individual Study	3	n/a	no
	CMPS	297B	Individual Study	1	n/a	no
Spring	CMPE	236	Understanding Cryptography	10	100	no
	CMPE	296	Masters Project	2	n/a	no
	CMPE	297A	Individual Study	4	n/a	no
	CMPE	297B	Individual Study	2	n/a	no

TEACHING 2018–19

				Enrolled	%Eval Retd	Shared?
Summer	CMPE	198	Individual Study or Research	1	n/a	no
	CMPS	299	Thesis Research	1	n/a	no
Fall	CMPE	280S	Seminar Computer Systems	8	50	no
	CMPE	297A	Individual Study	2	n/a	no
	CMPE	297B	Individual Study	2	n/a	no
	CMPE	299A	Thesis Research	1	n/a	no
	CMPS	111	Introduction to Operating Systems	44	23	no
	CMPE	195	Senior Thesis	4	n/a	no
Winter	CMPS	198	Individual Study or Research	1	n/a	no
	CMPS	297A	Individual Study	2	n/a	no
	CMPS	297C	Individual Study	1	n/a	no
	CMPS	299A	Thesis Research	1	n/a	no
	CMPS	280S	Seminar Computer Systems	11	36	no
	CMPE	296	Masters Project	1	n/a	no
	CMPE	297A	Individual Study	1	n/a	no
	CMPE	297B	Individual Study	1	n/a	no
	CMPE	299B	Thesis Research	1	n/a	no
	CMPE	299C	Thesis Research	1	n/a	no
Spring	CMPS	111	Introduction to Operating Systems	154	23	no

TEACHING 2019–2020

				Enrolled	%Eval Retd	Shared?
Fall	CSE	13S	Computer Systems and C Programming	49	18	no
	CSE	280S	Seminar Computer Systems	11	27	no
Winter	CSE	13S	Computer Systems and C Programming	90	11	no
	CSE	280S	Seminar Computer Systems	11	27	no
Spring	CSE	234	Understanding Cryptography	23	52	no
	CSE	280S	Seminar Computer Systems	6	17	no

Mentoring and Advising

Advisor to Continuing Students

Degree Goal	Year Entered	Name
M.S.	2022	Eugene Chou
M.S.	2020	Sabrina Au
Ph.D.	2020	Kyle Fredrickson
Ph.D.	2018	Austen Barker
Ph.D.	2016	Oceane Bel
Ph.D.	2015	James Hughes
Ph.D.	2010	Michael M ^c Throw

Postdoctoral Scholars

Name	Dates	Relationship	Degree Year	Institution
Yasu Ohara	2011 Fall–2013 Fall	Primary	2008	Keio University
Se Jin Kwon	2016 Spring–2017 Winter	Primary	2012	Ajou University

Doctor of Philosophy

Degree Year	Name	Relationship	Dissertation
2017	Joel Frank	Primary	Percival: A Reliable, Long-term, Distributed Storage System Free of Fixed-key Encryption
2017	Yan Li	Primary	ASCAR: Fully Autonomous Storage Contention Management System
2016	Stephanie Jones	Primary	Techniques for Reducing Long-term Data Movement on Shingled Magnetic Recording Drives
2016	Alexander Nelson	Primary	Software Signature Derivation From Sequential Digital Forensic Analysis
2016	Christina Strong	Primary	Data Allocation Approaches for Optimizing Storage Systems
2015	Alexandra Holloway	Primary	Design of a Data-Driven Micro-Display for Situation Awareness in Bursty Environments
2015	D. J. Capelis	Primary	Integrating Security into Lower Layers after Their Design
2013	Aleatha Parker-Wood	Primary	Improving File Management through Provenance and Rich Metadata
2010	Deepavali Bhagwat	Primary	Deduplication for Large Scale Backup and Archival Storage
2008	David Pease	Primary	Storage Tank: A Storage Area Network File System
2006	Lawrence You	Primary	Efficient Archival Data Storage
2005	Bo Hong	Co-Advisor	File and Storage Systems for MEMS-Based Storage
2002	Ahmed Amer	Primary	Predictive Data Grouping using Successor Prediction
2002	Tsozen Yeh	Primary	Improving File System Performance with File Access Predictions
2000	Theodore Haining	Primary	Non-volatile Cache Management for Improving Write Response Time with Rotating Magnetic Media
2000	Benjamin Reed	Primary	Using Network Attached Storage in a Secure Network Distributed File System
2000	Randal Burns	Primary	Data Management in a Distributed File System for Storage Area Networks
2000	Thomas Kroeger	Primary	Modeling File Access Patterns to Improve Caching Performance
1992	Richard Golding	Primary	Weak-consistency Group Communication and Membership

Doctor of Philosophy Dissertation Reading Committee

Degree Year	Name	Relationship	Advisor	Institution
—	Oceane Bel	Advisor	Prof. Darrell Long	UCSC
—	Devashish Purandare	Member	Prof. Ethan Miller	UCSC
—	James Hughes	Member	Prof. Darrell Long	UCSC
2019	Judith Samson	Member	Prof. Bradley Smith	UCSC
2019	Yuanjiang Ni	Member	Prof. Ethan Miller	UCSC
2019	Sinjoni Mukhopadhyay	Advisor	Prof. Darrell Long	UCSC
2019	James Byron	Member	Prof. Ethan Miller	UCSC
2019	Daniel Bittman	Member	Prof. Ethan Miller	UCSC
2015	Rekha Pitchumani	Member	Prof. Ethan Miller	UCSC
2015	D.J. Capelis	Member	Prof. Darrell Long	UCSC
2015	Alexandra Holloway	Advisor	Prof. Darrell Long	UCSC
2016	Raja Haddad	Président du jury	Witold Litwin	Université Paris–Dauphine
2015	Rekha Pitchumani	Member	Prof. Ethan Miller	UCSC
2014	Yangwook Kang	Member	Prof. Ethan Miller	UCSC
2014	Andrej Tolič	Member	Prof. Andrej Brodnik	University of Ljubljana
2013	Hanafi Yakouben	Membre du jury	Prof. Witold Litwin	Université Paris–Dauphine
2009	Karen Glocer	Member	Prof. Manfred Warmuth	UCSC
2009	Kevin Greenan	Member	Prof. Ethan Miller	UCSC
2009	Mark Storer	Member	Prof. Ethan Miller	UCSC
2007	Timothy Bisson	Member	Prof. Scott Brandt	UCSC
2007	John Colby	Member	Prof. Geoffrey Pullum	UCSC
2007	Dean Hildebrand	Member	Prof. Peter Honeyman	University of Michigan
2006	Zachary Peterson	Member	Prof. Randal Burns	Johns Hopkins University
2006	Feng Wang	Member	Prof. Scott Brandt	UCSC
2006	Caixue Lin	Member	Prof. Scott Brandt	UCSC
2005	Reiner Kraft	Member	Prof. Raymie Stata	UCSC
2005	Qin Xin	Member	Prof. Ethan Miller	UCSC
2005	Scott Banachowski	Member	Prof. Scott Brandt	UCSC
2004	Ismail Ari	Member	Prof. Ethan Miller	UCSC
2004	Dmitrii Zagorodnov	Member	Prof. Keith Marzullo	UCSD
2003	R. Kameshwari	Member	Prof. K. Gopinath	IIS Bangalore
2003	Bradley Smith	Member	Prof. J.J. Garcia-Luna	UCSC
2002	Brian Chess	Member	Prof. F. Joel Ferguson	UCSC
2002	Jyoti Raju	Member	Prof. J.J. Garcia-Luna	UCSC
1999	Rodrigo Garcés	Member	Prof. J.J. Garcia-Luna	UCSC
1999	Clay Shields	Member	Prof. J.J. Garcia-Luna	UCSC
1998	Bruce Montague	Member	Prof. Charles M ^c Dowell	UCSC
1996	Shree N. Murthy	Member	Prof. J.J. Garcia-Luna	UCSC
1996	Eric Rosen	Member	Prof. Patrick Mantey	UCSC
1995	Yuxia Zhang	Member	Prof. J.J. Garcia-Luna	UCSC
1993	Daniel Edelson	Member	Prof. Ira Pohl	UCSC
1993	Max Copperman	Member	Prof. Charles M ^c Dowell	UCSC

Doctor of Philosophy Qualifying Committee

Year	Name	Relationship	Advisor	Institution
2020	Shreyas Kumar	Member	Prof. Darrell Long	UCSC

2020	Devashish Purandare	Chair	Prof. Ethan Miller	UCSC
2020	James Hughes	Chair	Prof. Darrell Long	UCSC
2018	Judith Samson	Member	Prof. Bradley Smith	UCSC
2018	Trevor Pesout	Outside Member	Prof. Darrell Long	UCSC
2018	Yuanjing Ni	Chair	Prof. Ethan Miller	UCSC
2018	Sinjoni Mukhopadhyay	Member	Prof. Darrell Long	UCSC
2018	James Byron	Chair	Prof. Ethan Miller	UCSC
2018	Daniel Bittman	Chair	Prof. Ethan Miller	UCSC
2018	Oceane Bel	Member	Prof. Darrell Long	UCSC
2014	Preeti Gupta	Chair	Prof. Ethan Miller	UCSC
2012	Yangwook Kang	Chair	Prof. Ethan Miller	UCSC
2012	Rekha Pitchumani	Chair	Prof. Ethan Miller	UCSC
2012	Ian Adams	Chair	Prof. Ethan Miller	UCSC
2011	Avani Wildani	Chair	Prof. Ethan Miller	UCSC
2008	Neerja Bhatnagar	Chair	Prof. Ethan Miller	UCSC
2008	Kevin Greenan	Chair	Prof. Ethan Miller	UCSC
2008	Andrew Leung	Chair	Prof. Ethan Miller	UCSC
2007	Purvi Shah	Member	Prof. Jehan-François Pâris	University of Houston
2007	Mark Storer	Chair	Prof. Ethan Miller	UCSC
2006	David Essary	Member	Prof. Ahmed Amer	University of Pittsburgh
2005	Karen Glocer	Member	Prof. Manfred Warmuth	UCSC
2004	Reiner Kraft	Chair	Prof. Raymie Stata	UCSC
2003	Scott Banachowski	Chair	Prof. Scott Brandt	UCSC
2003	Qin Xin	Chair	Prof. Ethan Miller	UCSC
2003	Lawrence You	Member	Prof. Darrell Long	UCSC
2003	Feng Wang	Chair	Prof. Scott Brandt	UCSC
2003	Dmitrii Zagorodnov	Member	Prof. Keith Marzullo	UCSD
2003	Bo Hong	Member	Prof. Darrell Long	UCSC
2002	Ismail Ari	Chair	Prof. Ethan Miller	UCSC
2002	Bradley Smith	Chair	Prof. J.J. Garcia-Luna	UCSC
2001	Tsozen Yeh	Member	Prof. Darrell Long	UCSC
2001	Aman Shaikh	Member	Prof. Anujan Varma	UCSC
2000	Brian Chess	Chair	Prof. F. Joel Ferguson	UCSC
2000	Ahmed Amer	Member	Prof. Darrell Long	UCSC
1999	Srinivas Vutukury	Member	Prof. J.J. Garcia-Luna	UCSC
1999	Jyoti Raju	Member	Prof. J.J. Garcia-Luna	UCSC
1998	Marcelo Spohn	Member	Prof. J.J. Garcia-Luna	UCSC
1998	Gil Fuchs	Member	Prof. Robert Levinson	UCSC
1998	Benjamin Reed	Member	Prof. Darrell Long	UCSC
1997	Thomas Kroeger	Member	Prof. Darrell Long	UCSC
1997	Thomas Clay Shields	Member	Prof. J.J. Garcia-Luna	UCSC
1997	Steven Carter	Member	Prof. Darrell Long	UCSC
1997	Brian Levine	Member	Prof. J.J. Garcia-Luna	UCSC
1997	Brian Chess	Member	Prof. F. Joel Ferguson	UCSC
1996	Lampros Kalampoukas	Member	Prof. J.J. Garcia-Luna	UCSC
1996	Hans-Peter Dommel	Member	Prof. J.J. Garcia-Luna	UCSC
1996	Fumiaki Kamiya	Member	Prof. Allen Van Gelder	UCSC
1996	Dirk Coldeway	Member	Prof. Charles M ^c Dowell	UCSC
1996	Bruce Montague	Member	Prof. Charles M ^c Dowell	UCSC
1995	Theodore Haining	Member	Prof. Darrell Long	UCSC
1995	Shree N. Murthy	Member	Prof. J.J. Garcia-Luna	UCSC

1995	Jochen Behrens	Member	Prof. J.J. Garcia-Luna	UCSC
1995	Dimitrios Stiliadis	Member	Prof. Anujan Varma	UCSC
1995	Chane Fullmer	Member	Prof. J.J. Garcia-Luna	UCSC
1994	Gil Fuchs	Member	Prof. Robert Levinson	UCSC
—	Tsozen Yeh	Member	Prof. Darrell Long	UCSC
—	Thomas Shields	Member	Prof. J.J. Garcia-Luna	UCSC
—	Steven Carter	Member	Prof. Darrell Long	UCSC
—	Shree Murthy	Member	Prof. J.J. Garcia-Luna	UCSC
—	Sage Weil	Member	Prof. Scott Brandt	UCSC
—	Richard Golding	Member	Prof. Darrell Long	UCSC
—	Randal Burns	Member	Prof. Darrell Long	UCSC
—	Max Copperman	Member	Prof. Charles M ^c Dowell	UCSC
—	Marcelo Spohn	Member	Prof. J.J. Garcia-Luna	UCSC
—	Lawrence You	Member	Prof. Darrell Long	UCSC
—	Kjell Post	Member	Prof. Allen Van Gelder	UCSC
—	Ignacio Solis	Member	Prof. J.J. Garcia-Luna	UCSC
—	Ignacio Corderi	Member	Prof. Darrell Long	UCSC
—	Gil Fuchs	Member	Prof. Robert Levinson	UCSC
—	Fumiaki Okushi	Member	Prof. Allen Van Gelder	UCSC
—	Eric Rosen	Member	Prof. Patrick Mantey	UCSC
—	Dirk Coldewey	Member	Prof. Charles M ^c Dowell	UCSC
—	Dimitrios Stiliadis	Member	Prof. J.J. Garcia-Luna	UCSC
—	David Pease	Member	Prof. Darrell Long	UCSC
—	Chane Fullmer	Member	Prof. J.J. Garcia-Luna	UCSC
—	Brian Levine	Member	Prof. J.J. Garcia-Luna	UCSC
—	Bo Hong	Member	Prof. Scott Brandt	UCSC
—	Benjamin Reed	Member	Prof. Darrell Long	UCSC
—	Aman Shaikh	Member	Prof. J.J. Garcia-Luna	UCSC
—	Ahmed Amer	Member	Prof. Darrell Long	UCSC

Master of Science

Degree Year	Name	Relationship	Thesis
2015	Ignacio Corderi	Primary	RESAR: A System for a Two-Failure Tolerant, Self-Adjusting Million Disk Storage Cluster
2014	Erik Steggall	Primary	Sleuthkit: XTAF extension
2014	Michael M ^c Throw	Primary	Extended Metadata and File System Design: Designing File Systems that Take Advantage of Rich Metadata
2013	Sonali Somyalipi	Primary	Impact of Differential Compression and Deduplication on Genomic Data Storage
2012	Akhilesh Bhargav Malavalli	Primary	Improved Measurement of Backup Reconstruction Performance for a Deduplicated System
2010	Stephanie Jones	Primary	On-line Deduplication in a Log-Structured File System for Primary Storage
2008	Rosie Wacha	Primary	Data Reliability Techniques for Specialized Storage Environments
2007	Kristal Pollack	Primary	Quota Enforcement for High-Performance Distributed Storage Systems
2005	Jeffrey Rybczynski	Primary	Active Workload Reshaping to Maximize Disk Energy Conservation
2005	Jörg Meyer	Primary	Large Scale Multi-Type Inverted List Indexing
2004	Svetlana Kagan	Primary	Using Containers for Efficient Storage of Small Files in the Deep Store Archival System
2004	Karl Brandt	Primary	Using Multiple Experts to Perform File Prediction
2003	Ian Brown	Primary	Upgrading NFS File Servers in a Live Environment
2003	Karen Glocer	Primary	The Utility of Fine-Grained Workload Characterizations
2002	Ying Lin	Primary	Power Conservation Strategies for MEMS-Based Storage Devices
2002	Zachary Peterson	Primary	Data Placement for Copy-on-Write Using Virtual Contiguity
2000	Pu Yang	Primary	Modeling Probe Based Data Storage Devices
2000	A. David M ^c Nab	Primary	Extensible UNIX Access Control Lists
1999	John Colby	Primary	Error Correction for Incremental LR Parsers
1998	Tracey Sconyers	Primary	An Examination of Power Consumption in Wireless Modems
1997	Randal Burns	Primary	Differential Compression: A Generalized Solution for Binary Files
1996	Thomas Kroeger	Primary	Predicting File System Actions From Reference Patterns
1994	David Schreiber	Primary	A Code Generator for the PowerPC 601
1994	Chane Fullmer	Primary	Implementation of Adaptive Flow Control Mechanism to the Swift Redundant Distributed File Architecture
1993	Michelle Abram	Primary	Transparent Remote Procedure Calls
1993	Kulumani B. Sriram	Primary	A Study of the Reliability of Hosts on the Internet
1993	Cheng Tang	Primary	A Transport Mechanism for Supporting Multimedia Traffic in ATM Networks
1992	William Osser	Primary	Automatic Process Selection for Load Balancing
1992	Carol Osterbrock	Primary	A Transport Level Protocol for Guaranteed Performance in a Local Area Network
1991	Richard Golding	Primary	Accessing Replicated Data in a Large Scale Distributed System

Master of Science Thesis Reading Committee

Year	Name	Relationship	Advisor
2018	Isaak Cherdak	Chair	Prof. Ethan Miller
2018	Matthew Bryson	Member	Prof. Ethan Miller
2018	Oceane Bel	Chair	Prof. Darrell Long
2018	Austen Barker	Chair	Prof. Darrell Long
2017	Preeti Gupta	Chair	Prof. Darrell Long
2015	Xiaoyuan Lu	Member	Prof. Darrell Long
2015	Ignacio Corderi	Chair	Prof. Darrell Long
2011	Nathan Edel	Member	Prof. Ethan Miller
2010	Kiren Jin	Member	Prof. Ethan Miller
2004	Narayan Brooks	Member	Prof. Suresh Lodha
2003	David Pease	Member	Prof. Richard Hughey
2002	Reiner Kraft	Member	Prof. Raymie Stata
2002	Feng Wang	Member	Prof. Scott Brandt
2002	Alicia Szczurowska	Member	Prof. Scott Brandt
2000	Paul Caamano	Member	Prof. Charles M ^c Dowell
1999	Manosiz Bhattacharrya	Member	Prof. Charles M ^c Dowell
1998	Christina Parsa	Member	Prof. J.J. Garcia-Luna
1998	Jyoti Raju	Member	Prof. J.J. Garcia-Luna
1997	Bradley Smith	Member	Prof. J.J. Garcia-Luna
1997	Michael Allen	Member	Prof. Charles M ^c Dowell
1997	Bruce Sherrod	Member	Prof. David Helmbold
1996	Brian Levine	Member	Prof. J.J. Garcia-Luna
1996	Thomas Clay Shields	Member	Prof. J.J. Garcia-Luna
1996	Catherine Wilson	Member	Prof. Suresh Lodha
1995	Brian Chess	Member	Prof. F. Joel Ferguson
1994	Shree N. Murthy	Member	Prof. J.J. Garcia-Luna
1993	Theodore Haining	Member	Prof. Charles M ^c Dowell
1993	Rock Pfothenhauer	Member	Prof. Ira Pohl
1993	Richard Snyder	Member	Prof. Robert Levinson
1991	Matthew Breden	Member	Prof. Tracy Larrabee
1990	Max Copperman	Member	Prof. Charles M ^c Dowell
1990	Daniel Edelson	Member	Prof. Ira Pohl
1990	Frank Jas	Member	Prof. Karl Schimpf
1989	James Kerr	Member	Prof. Allen Van Gelder

Bachelor of Science

Degree Year	Name	Relationship	Thesis
2019	Yash Gupta	Primary	Artifice: A Deniable Steganographic File System
2016	Daniel Bittman	Primary	Designing a Wait-free MPSC Queue
2011	Sulimon Sattari	Primary	A Cross-Disciplinary Data Sharing and Querying Platform for Earth Science Data
2001	Christopher Benson	Primary	Peer-to-Peer File System using Weak Consistency Replication
1994	Sean Gilligan	Primary	MacMet: A Macintosh Program to Display Real-Time and Historical Data from Meteorological Stations
1992	Nitin Ganatra	Primary	Census: Collecting Host Information on a Wide Area Network
1991	Aaron Emigh	Primary	The Swift Architecture: Anatomy of a Prototype
1989	Leland Wallace	Primary	Font Extensions to the X-Fig Graphics Program