David Glasco, Ph.D.

10308 Tularosa Pass
Austin, TX 78726
(512) 577-8693
daveglasco@swbell.net
www.davidglasco.com

Professional Summary

Twenty-five years of technical leadership across a wide range of technologies with a strong emphasis on driving innovative computing architectures and platforms in both industrial and academic environments. Versatile background including leadership, computer architecture and full-cycle product design. Experience ranging from academic and industrial research to pre-IPO startups to world-class, Fortune 500 corporations.

Skills

Computer Architecture: Developed architectures and micro-architectures for multiple high performance systems including high-end, multi-core GPUs, multi-processor ccNUMA systems, high-end processor designs; mobile devices, image processing pipelines, computer vision, wafer-scale systolic arrays, and multiprocessor performance data acquisition logic. Strong focus on system-on-a-chip (SOC) architectures, network-on-chip (NoC) architectures, cache hierarchies, cache coherence protocols, and front-end memory controllers.

Leadership: Built multiple research and development teams with strong focus on culture and collaborative accomplishment. Coordinated and provided leadership for several cross-functional and informal teams. Involved in many aspects of strategic product development including customer engagement, market analysis and product requirement definitions. Drove future roadmaps for several efforts. Developed IP strategic plan to maximize patent protection.

Design & Modeling: Significant design experience ranging from system prototyping to logic design and verification. Strong chip design experience including micro-architecture, RTL development, logic verification and behavioral modeling. Experience with high-end FPGAs, ASIC and custom logic design for high-performance computing, computer graphics, image processing, radar anti-jamming, networking, and security systems. Developed and brought-up several system prototypes to demonstrate and analyzed system behavior and performance. Developed modeling environments for system, protocol and micro-architecture performance analysis and utilized environments for detailed analysis of latency, throughput, resource sizing, algorithms, etc. Several projects required extensive software development: driver stacks, system software, simulators, user interfaces, data acquisition and display.

Industrial Experience

NVIDIA

Director of Architecture, Computational Imaging Group 2013-Present

Leading both the architecture and RTL teams for NVIDIA's computational imaging pipeline (camera), which is responsible for the processing and enhancements of raw sensor images in the Tegra camera pipeline. Developing a hardware pipeline (Image Signal Processor) to convert raw Bayer sensor data to high-quality images and video. Focusing on developing new innovative imaging features and supporting SOC system architectures and designs. Working to improve the 3As (auto-focus, auto-white balance, auto exposure), noise reduction and image quality.

Senior Research Manager, NVIDIA Research 2011-2013

Responsible for a cross-functional research team designing GPU-based SOC architectures for future systems ranging from high-performance computing to graphics to mobile devices. Focusing on a broad application space – graphics, bioinformatics, web servers, computer vision, machine learning, databases, simulations, etc. Built a cross-functional, matrixed team to focus on all aspects of the SOC system design – applications, architecture, micro-architecture, implementation, circuit technology, compilers, system software, and more. Successfully designed and executed a product transition plan in which both NV Research ideas and staff were transferred to the product groups.

Senior Architecture Manager, GPU System Architecture 2008-2011

Responsible for the architectural development of a significant portion of NVIDIA's Kepler generation of GPU System-On-Chip (SOC) system architecture focusing on the memory system. This work included the on-chip interconnect (network-on-chip), memory controller, memory caches, virtual memory, memory access protocols, system interface, and multi-chip interconnects. Explored CPU/GPU interactions and supporting memory structures including hardware-based coherent cache architectures. Responsible for the architecture, micro-architecture, performance modeling and validation, functional modeling and validation of the GPU memory system. Responsibilities included the current GPU development, next generation GPU development and the roadmap development for future GPU memory systems. Managing multiple GPUs in development each with multiple teams, which were spread across multiple sites. Close interactions with hardware and physical design teams.

Architecture Unit Lead, GPU Architecture 2004-2008

Architecture unit lead responsible for the development of the memory system including the on-chip interconnect (NoC), memory protocols, memory cache and virtual memory system of NVIDIA's Fermi generation of GPU SOCs. Responsibilities include the development of the architecture, micro-architecture, performance modeling and validation, and functional modeling and validation. Drove multiple performance analysis efforts that impacted full-chip performance. Also responsible for the unit management – plans, schedule, staffing, assignments, etc.

Architect, GPU Architecture 2003 - 2004

Responsible for the development of the virtual memory architecture for NVIDIA's GPU in support of Microsoft's Longhorn Device Driver Model (LDDM). Contributed to many other aspects of architecture work on NVIDA GPUs. Contributed to and later drove NVIDIA's involvement in Microsoft's Virtualized Graphics effort.

Newisys, Inc. Austin, TX

Chief Architect, Chip Development 2000 - 2003

Responsible for the overall architectural and micro-architectural development of Newisys' CC-NUMA cache coherence controller and low-latency packet switch for scalable multiprocessor systems based on AMD's Opteron x86-64 processors (Horus). Architected the overall system design, coherence controller design and coherence protocol. Co-developed the design's micro-architecture. Responsible for the extended HyperTransport protocol development, which included coherence directory and remote data cache functionality, and subsequent behavioral modeling. Drove the development of a cycle-accurate performance model and subsequent analysis. Worked closely with the BIOS and service processor development teams.

Drove the development of an advanced FPGA prototyping environment for analysis and validation of the multiprocessor system. Built the prototype on top of existing 2-processor Opteron systems. Effort included cross-functional technical leadership, system design, board design, software development, logic design, logic synthesis and place & route. Utilized high-end Xilinx FPGAs to prototype controller design.

Developed and drove the Newisys intellectual property development for the scalable multiprocessor systems. Built a strong patent portfolio (40+ patents). Developed blocking IP strategy and reviewed strategy with two external IP law firms.

Technical Manager, Architecture Group/Chip Design Group 2000 - 2003

Built and managed two groups within Newisys: initially the chip design group and finally the architecture group. Chip design group responsibilities included project plan development, interviewing and hiring, culture development, micro-architecture and initial RTL development. Built an advanced architecture group once the design group was stable. Architecture group responsibilities include performance modeling and analysis; prototype development, validation and behavior analysis; cache coherence protocol development and analysis; and future product development.

Intel/Texas Development Center, Desktop Products Group Austin, TX

Architect/Technical Manager, CPU System Cluster 1999 - 2000

Technical manager responsible for building and leading an engineering team that was responsible for the system components of a high performance, IA-32 processor with integrated memory controller and micro-architectural support for multiple, heterogeneous on-die cores. Provided technical leadership for the team's efforts, which included multiprocessor cache coherence protocol architecture and development; protocol

engine micro-architecture and implementation; memory controller micro-architecture and implementation; protocol formal verification; and system level performance modeling and analysis.

IBM Research/Austin Research Laboratory Austin, TX

Architect/Research Staff Member 1996 - 1999

Provided technical leadership for a small research team that successfully implemented and demonstrated IBM's first Intel-based CC-NUMA hardware prototype. The team architected, designed and implemented the system using a combination of off-the-shelf components and programmable logic. Functionality included a patented hardware performance monitor to understand system performance. Worked closely with the CC-NUMA software team to understand system performance and drive performance monitoring and enhancements into Windows NT and SCO UnixWare. See "Experience with building a commodity Intel-based ccNUMA system" and "Windows NT in a CC-NUMA System."

Architected, designed and implemented the cache coherence mechanism for IBM's first functional PowerPC-based CC-NUMA hardware prototype. Work included the development of the cache coherence protocol and the micro-architecture & implementation of the coherence directory and pending request mechanism. Co-architected the overall CC-NUMA adapter. Demonstrated hardware functionality of a three-node CC-NUMA system implemented using high-speed programmable logic (FPGA). Developed several innovative and patented protocol features to overcome deficiencies in the PowerPC bus architecture.

HP Laboratories Palo Alto, CA

Architect/Member Technical Staff 1995-1996

Investigated cache coherence structures for HP's future generation shared-memory multiprocessor systems. Focused on cache coherence protocols, directory structures and the interaction between the cache coherence protocol, processor cache hierarchy, operating system and application software. Investigated other memory latency tolerating and reducing techniques to give HP an advantage over its competition.

HaL Computer Systems Campbell, CA

Architect/Verification Engineer 1994 - 1995

Developed a verification strategy, which was based on high-level modeling (HLM), for a CC-NUMA system. The strategy included formal verification (FV) of the cache coherence protocol and a verification tool that was able to compare the results of cycle and non-cycle accurate models. Implemented portions of the HLM using verilog and developed early FV models of the protocol. Worked with Prof. Dill of Stanford to improve the FV tool and methodology – funding one graduate student.

MIT Lincoln Laboratory Boston, MA

Micro-Architect & Logic Design Engineer 1988 - 1990

Architected a radar adaptive nulling hardware prototype designed around a systolic array, which was constructed from an array of custom CORDIC data processors. Effort included the design of a high-speed dual banked memory system and data path control logic for the systolic array. Additionally, the effort included system design, board design, and discrete and programmable logic. System included a micro-controller that required extensive programming. Significant software was also developed on the host systems to feed data to the systolic array, analyze output data, and present results. Successfully demonstrated the system in both an IBM PC and Sun workstation environment.

Artisoft, Inc Tucson, AZ

Logic Design Engineer 1985 - 1987

While attending U of Arizona, worked as a part time engineer and developed several products for the IBM PC including a hardware access control card, a laptop to desktop networking system software and portions of a local area network card and software. Involved in all aspects of product development including product conception, logic design, implementation, board design, debug, verification, manufacturing and marketing.

Academic Experience

University of Texas at Austin — Austin, TX

Ph.D. Committee, EE Department — 2001-2003

Participating in the orals committee for a Ph.D. student in the Electrical Engineering department. Student's work is focused on high-end processor design with an emphasis on power-wise design.

Stanford University — Stanford, CA

Consulting Assistant Professor, EE Department — 1995 - 1999

Consulting Professor working with Professor Michael Flynn. Developed and taught a graduate level course on shared-memory multiprocessors. Obtained an industrial grant to fund research in fault-tolerant multiprocessors. Actively participated in the research and advised a graduate student funded by this grant. Graduated one Ph.D. student.

Stanford University — Stanford, CA

Research Assistant, Ph.D. Degree Program, EE Department — 1990 - 1994

Designed update-based cache coherence protocols for scalable shared-memory multiprocessors. Designed protocols for both distributed and centralized directory structures. Developed a set of architectural models for shared-memory multiprocessors and several shared-memory applications. Analyzed the performance of the update-based protocols with respect to common invalidate-based protocols through full system simulations. Identified protocol limitations and evaluated possible protocol enhancements to overcome these limitations. Formally verified the update-based protocols using the Murphi modeling checking tool from Stanford.

Education

Stanford University – Center for Professional Development — Stanford, CA

Bioinformatics Graduate Certificate, complete 2 of 3 required courses, GPA 4.0/4.0 — In progress

Computational Molecular Biology (BIOC218)
Computational Structural Biology (SBIO228)

University of Texas — Austin, TX

The Institute for Managerial Leadership for Engineers, Scientist and Computer Professionals Certificate Program, The Texas Business School — 1998 - 1999

Stanford University — Stanford, CA

Ph.D., Electrical Engineering, GPA 3.9/4.0 — 1991 - 1994

Dissertation: "Design and Analysis of Update-Based Cache Coherence Protocols for Scalable Shared-Memory Multiprocessors"
Advisor: Professor Michael J. Flynn

M.S., Electrical Engineering, GPA 3.9/4.0 — 1990 - 1991

University of Arizona — Tucson, AZ

B.S., Computer Engineering, GPA 3.9/4.0, Magna Cum Laude — 1984 – 1987

Scuba Schools International (SSI) — Austin, TX

Dive Control Specialist Instructor — 2007-present

Patents

Issued (85)

NVIDIA

8,607,008	System and method for independent invalidation on a per engine basis
8,601,235	System and method for concurrently managing memory access requests
8,595,437	Compression status bit cache with deterministic isochronous latency
8,543,792	Memory access techniques including coalescing page table entries
8,539,130	Virtual channels for effective packet transfer
8,504,794	Override system and method for memory access management
8,504,773	Storing dynamically sized buffers within a cache
8,464,001	Cache and associated method with frame buffer managed dirty data pull and high-priority clean mechanism
8,392,667	Deadlock avoidance by marking CPU traffic as special
8,359,454	Memory access techniques providing for override of page table attributes
8,352,709	Direct memory access techniques that include caching segmentation data
8,347,065	System and method for concurrently managing memory access requests
8,347,064	Memory access techniques in an aperture mapped memory space
8,327,071	Interprocessor direct cache writes
8,325,194	Mitigating main crossbar load using dedicated connections for certain traffic types
8,271,734	Method and system for converting data formats using a shared cache coupled between clients and an external memory
8,244,984	System and method for cleaning dirty data in an intermediate cache using a data class dependent eviction policy
8,234,478	Using a data cache array as a DRAM load/store buffer
8,234,458	System and method for maintaining cache coherency across a serial interface bus using a snoop request and complete message
8,185,602	Transaction processing using multiple protocol engines in systems having multiple multi-processor clusters
8,156,404	L2 ECC implementation
8,135,926	Cache-based control of atomic operations in conjunction with an external ALU block
8,131,931	Configurable cache occupancy policy
8,108,610	Cache-based control of atomic operations in conjunction with an external ALU block
8,099,650	L2 ECC implementation
8,065,465	Mitigating main crossbar load using dedicated connections for certain traffic types
8,060,700	System, method and frame buffer logic for evicting dirty data from a cache using counters and data types
7,859,541	Apparatus, system, and method for using page table entries in a graphics system to provide storage format information for address translation
7,769,979	Caching of page access parameters
7,382,377	Render to texture cull

Newisys

8,572,206	Transaction processing using multiple protocol engines
7,653,790	Methods and apparatus for responding to a request cluster
7,577,755	Methods and apparatus for distributing system management signals
7,577,727	Dynamic multiple cluster system reconfiguration
7,545,382	Apparatus, system, and method for using page table entries in a graphics system to provide storage format information for address translation
7,418,517	Methods and apparatus for distributing system management signals

7,395,379	Methods and apparatus for responding to a request cluster
7,395,347	Communication between and within multi-processor clusters of multi-cluster computer systems
7,386,626	Bandwidth, framing and error detection in communications between multi-processor clusters of multi-cluster computer systems
7,346,744	Methods and apparatus for maintaining remote cluster state information
7,337,279	Methods and apparatus for sending targeted probes
7,334,089	Methods and apparatus for providing cache state information
7,296,121	Reducing probe traffic in multiprocessor systems
7,281,055	Routing mechanisms in systems having multiple multi-processor clusters
7,272,688	Methods and apparatus for providing cache state information
7,251,698	Address space management in systems having multiple multi-processor clusters
7,249,224	Methods and apparatus for providing early responses from a remote data cache
7,222,262	Methods and devices for injecting commands in systems having multiple multi-processor clusters
7,162,589	Methods and apparatus for canceling a memory data fetch
7,159,137	Synchronized communication between multi-processor clusters of multi-cluster computer systems
7,155,525	Transaction management in systems having multiple multi-processor clusters
7,117,419	Reliable communication between multi-processor clusters of multi-cluster computer systems
7,107,409	Methods and apparatus for speculative probing at a request cluster
7,107,408	Methods and apparatus for speculative probing with early completion and early request
7,103,823	Communication between multi-processor clusters of multi-cluster computer systems
7,103,726	Methods and apparatus for managing probe requests
7,103,725	Methods and apparatus for speculative probing with early completion and delayed request
7,103,636	Methods and apparatus for speculative probing of a remote cluster
7,069,392	Methods and apparatus for extended packet communications between multiprocessor clusters
7,047,372	Managing I/O accesses in multiprocessor systems
7,039,740	Interrupt handling in systems having multiple multi-processor clusters
7,024,521	Managing sparse directory evictions in multiprocessor systems via memory locking
7,003,633	Methods and apparatus for managing probe requests
6,950,913	Methods and apparatus for multiple cluster locking

IBM Research

6,934,814	Cache coherence directory eviction mechanisms in multiprocessor systems which maintain transaction ordering
6,925,536	Cache coherence directory eviction mechanisms for unmodified copies of memory lines in multiprocessor systems
6,920,532	Cache coherence directory eviction mechanisms for modified copies of memory lines in multiprocessor systems
6,865,595	Methods and apparatus for speculative probing of a remote cluster
6,499,028	Efficient identification of candidate pages and dynamic response in a NUMA computer
6,421,775	Interconnected processing nodes configurable as at least one non-uniform memory access (NUMA) data processing system
6,349,394	Performance monitoring in a NUMA computer
6,279,085	Method and system for avoiding livelocks due to colliding writebacks within a non-uniform memory access system
6,275,907	Reservation management in a non-uniform memory access (NUMA) data processing system
6,269,428	Method and system for avoiding livelocks due to colliding invalidating transactions within a non-uniform memory access system

6,266,743	Method and system for providing an eviction protocol within a non-uniform memory access system
6,226,718	Method and system for avoiding livelocks due to stale exclusive/modified directory entries within a non-uniform access system
6,192,452	Method and system for avoiding data loss due to cancelled transactions within a non-uniform memory access system
6,178,472	Queue having distributed multiplexing logic
6,148,361	Interrupt architecture for a non-uniform memory access (NUMA) data processing system
6,145,032	System for recirculation of communication transactions in data processing in the event of communication stall
6,115,804	Non-uniform memory access (NUMA) data processing system that permits multiple caches to concurrently hold data in a recent state from which data can be sourced by shared intervention
6,085,293	Non-uniform memory access (NUMA) data processing system that decreases latency by expediting rerun requests
6,081,874	Non-uniform memory access (NUMA) data processing system that speculatively issues requests on a node interconnect
6,067,611	Non-uniform memory access (NUMA) data processing system that buffers potential third node transactions to decrease communication latency
6,067,603	Non-uniform memory access (NUMA) data processing system that speculatively issues requests on a node interconnect

Publications

Conference/Journal Papers

Keckler, S.W.; Dally, W.J.; Khailany, B.; Garland, M.; Glasco, D., "GPUs and the Future of Parallel Computing," in IEEE Micro, Volume: 31, Issue: 5, pages 7-17, 2011

Brock, B. C., Carpenter, G. D., Chiprout, E., Dean, M. E., De Backer, P. L., Elnozahy, E. N., Franke, H., Giampapa, M. E., Glasco, D. B., Peterson, J. L., Rajamony, R., Ravindran, R., Rawson, F. L., Rockhold, R. L., and Rubio, J. "Experience with building a commodity Intel-based ccNUMA system," in IBM Journal of Research and Development, pages 207-228,Volume 45, Number 2, 2001.

Brock, B. C., Carpenter, G. D., Chiprout, E., Elnozahy, E. N., Dean, M., Glasco, D. B., Peterson, J. L., Rajamony, R., Rawson, F. L., Rockhold, R. L. and Zimmerman, A. "Windows NT in a CC-NUMA System," *Proceedings of the 3rd USENIX Windows NT Symposium, Seattle, Washington,* pages 61-72, July 1999.

Sunada, D., Glasco, D. B. and Flynn, M. J. "Multiprocessor Architecture Using an Audit Trail for Fault Tolerance," [PDF] In *Proceedings of the Twenty-Ninth Annual International Symposium on Fault-Tolerant Computing*, pages 40-47, June 1999.

Sunada, D., Glasco, D. B. and Flynn, M. J. "ABSS v2.0: a SPARC Simulator," In *Proceedings of the Eighth Workshop on Synthesis and System Integration of Mixed Technologies*, pages 143-149, October 1998.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Using Fine-Grain Data Synchronization on Scalable Shared-Memory Multiprocessors," In *Proceedings of the International Conference of Parallel Architectures and Compilation Techniques*, pages 79-88, August 1994.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Write Grouping for Update-Based Cache Coherence Protocols," In *Proceedings of the Sixth IEEE Symposium of Parallel and Distributed Processing*, pages 334-341, October 1994.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Update-Based Cache Coherence Protocols for Scalable Shared-Memory Multiprocessors," In *Proceedings of the Twenty-seventh Annual Hawaii International Conference of System Sciences*, pages 534-545, January 1994.

Glasco, D. B. and Sargent, M., "Using IBM's Marvelous Keyboard," *Byte Magazine*, pages 402-415, May 1983.

Book Chapters

Contributed to chapter 8, "Shared Memory Multiprocessors" of *Computer Architecture: Pipelined and Parallel Processor Design*, Michael J. Flynn, Jones and Bartlett Publishers, Boston, 1995.

Contributed to chapter 8, "Keyboard and Video Display" of The *IBM Personal Computer from the Inside Out*, Murry Sargent, III and Richard L. Shoemaker, Addison-Wesley Publishing Company, 1984.

Technical Reports

Sunada, D., Glasco, D. B. and Flynn, M. J. "Novel Checkpointing Algorithm for Fault Tolerance on a Tightly-Coupled Multiprocessor," Technical Report CSL-TR-99-776, Computer Systems Laboratory, Stanford University, January 1999.

Sunada, D., Glasco, D. B. and Flynn, M. J. "Hardware-Assisted Algorithms for Checkpoints," Technical Report CSL-TR-98-756, Computer Systems Laboratory, Stanford University, June 1998.

Sunada, D., Glasco, D. B. and Flynn, M. J. "ABSS v2.0: a SPARC Simulator," Technical Report CSL-TR-98-755, Computer Systems Laboratory, Stanford University, April 1998.

Sunada, D. Glasco, D. B. and Flynn, M. J. "Fault Tolerance: Methods of Rollback Recovery," Technical Report CSL-TR-97-718. Computer Systems Laboratory, Stanford University, March 1997.

Glasco, D. B. "Analysis of Update-Based Cache Coherence Protocols for Scalable Shared-Memory Multiprocessors," Ph.D. Dissertation, Computer Systems Laboratory, Stanford University, December 1994. Also appears as Technical Report CSL-TR-94-670, Computer Systems Laboratory, Stanford University, June 1995.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Write Grouping for Update-Based Cache Coherence Protocols," [PDF] Technical Report CSL-TR-94-612, Computer Systems Laboratory, Stanford University, March 1994.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "The Impact of Cache Coherence Protocols On Systems Using Fine-Grain Data Synchronization," Technical Report CSL-TR-94-611, Computer Systems Laboratory, Stanford University, March 1994.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Design and Validation of Update-Based Cache Coherence Protocols," Technical Report CSL-TR-94-613, Computer Systems Laboratory, Stanford University, March 1994.

Glasco, D. B., Delagi, B. A. and Flynn, M. J. "Update-Based Cache Coherence Protocols for Scalable Shared-Memory Multiprocessors," Technical Report CSL-TR-94-588, Computer Systems Laboratory, Stanford University, November 1993.

Rader, C. M., Allen, D. L., Glasco, D. B. and Woodward, C. E. "MUSE - A Systolic Array for Adaptive Nulling with 64 Degrees of Freedom, Using Givens Transformations and Wafer Scale Integration," Technical Report 886, Lincoln Laboratory, Massachusetts Institute of Technology, May 1990.

Research Disclosures

Glasco, D. B., Bannister, J. P., Carpenter, G. D., Dean, M. E., Dickol, J. E., Iachetta, R. N., Parades, J. A. and Rosilier, A. C. "sNUMA Cache Coherence Shared Memory Multiprocessor," IBM Research Disclosures, 1999.

Glasco, D. B. "Pipelined Directory Structure for CC-NUMA Systems," IBM Research Disclosures, 1999.

Glasco, D. B. "Distributed Pending Buffer for CC-NUMA Systems," IBM Research Disclosures, 1999.

Glasco, D. B. "Automatic Generation of Coherence Protocol Logic," IBM Research Disclosures, 1999.

Glasco, D. B. "Mirroring Data Modifications in a CC-NUMA System," IBM Research Disclosures, 1999.

Bannister, J. P., Carpenter, G. D., Dean, M. E., De Backer, P. L., Dickol, J. E., Glasco, D. B., Iachetta, R. N., Paredes, J. A. and Rockhold, R. L. "Startup Methodology for a NUMA Machine," IBM Research Disclosures, December 1998.

Bannister, J. P., Carpenter, G. D., Dean, M. E. and Glasco, D. B. "Non-Blocking Distributed Bus Switch for Multicomputer Systems," IBM Research Disclosures, August 1998.

Invited Presentations

"Topics in Multiprocessors," Presented at University of Texas, April 1998.

"Current Topics in Multiprocessors," Presented at Stanford University, February 1996.

"Update-Based Cache Coherence Protocols," Presented at IBM Research – ARL, January 1996.

"Update-Based Cache Coherence Protocols," Presented at Silicon Graphics Incorporated, March 1994.

"Update-Based Cache Coherence Protocols," Presented at NASA Ames Research Center, February 1994.