Advanced media-oriented systems research: ubiquitous capture, interpretation, and access

College of Computing
Georgia Institute of Technology

Annual Report via Fastlane July 20, 2003

Principal Investigators: *Umakishore Ramachandran, Mustaque Ahamad, Chris Atkeson, Ann Chervenak, and Karsten Schwan*

Contributing Investigators: *Gregory Abowd, Irfan Essa, Richard Fujimoto, Ken Mackenzie, Ed Omiecinski, and Sudhakar Yalamanchili*

Other Senior Personnel: *Sung Kyu Lim, Hsien-Hsin Lee, Ling Liu, Calton Pu, and Jim Rehg*

1 Activities and Findings

1.1 Research and Education

This project aims to provide end-to-end infrastructure for capture, interpretation and access of data streams in sensor-rich, pervasive computing environments. This subsection details the research accomplishments this past year (2002-03) as well as plans for the coming year (2003-04).

1.1.1 High Performance Distributed Applications

One major component of our efforts continues to be centered around developing technologies for interactive, high-performance, distributed applications. Our work in this area has been jointly funded by NSF and DOE. During the last year, we have (1) developed realistic applications jointly with end users, (2) created adaptive middleware technologies to better cope with dynamic changes in resources available to such applications, and (3) studied application execution and performance with adaptive middleware on representative testbeds (additional funding from HP and Intel).

The key problem addressed by our research is the dynamic nature of distributed hardware resources used by high performance applications coupled with the dynamic changes in end user needs. Interactive applications will be useful to end users only if certain quality of service (QoS) needs are continuously met, throughout an interactive session. One specific example studied by our group, jointly with researchers from DOE Oakridge, is an interactive molecular dynamics application, described in detail in [1] The novel nature of this application lies in its ability to dynamically customize end users' views of shared scientific data, whether such data is in storage, is being produced by a running simulation, or is derived from remote

instruments. In addition, end users can combine data from any number of remote sources, as well as display it on a variety of remote devices ranging from handhelds to immersive desks.

Creating and experimenting with this application has resulted in interesting technical outcomes. One outcome demonstrates the importance of coordinating the behavior of adaptive middleware with the behavior of network protocol stacks, in order to control the way in which data is streamed to end users. Without such coordination, the large data volumes transferred by scientific applications can result in high levels of jitter [2]. These results were attained with an Emulab-based network testbed at Georgia Tech, using some of the cluster machines acquired with RI funding. Results now being attained are using remote network links to DOE collaborators.

Another set of outcomes concerns the manner in which scientific data is represented, in order to permit it to be easily interchanged across scientific collaborators. We use XML as metadata markup, augmented with an efficient binary representation of scientific data, to provide data interoperability with reduced data size and transmission costs. This work is documented in detail in several publications [3, 4, 5, 6, 7, 8, 9].

Another set of techniques links network-level knowledge about current communication paths in mobile systems to middleware level actions, such as the use of appropriate overlay networks in mobile systems [10] These Java-based technologies mirror our ongoing C/C++-based work focused on high end applications. In such work, we are creating runtime techniques for binary code generation that deploy appropriate code modules, at runtime, onto the machines used by remote collaborators. Such code modules typically implement the dynamic data filtering techniques needed for remote sensing or visualization [11].

User-level middleware for distributed applications can provide certain levels of quality of service to end users. However, without kernel-level support, QoS guarantees are hard to provide, and resource management is inefficient or difficult to perform. Substantial work by our group has created kernel-level resource management mechanisms, for both mobile and distributed embedded systems and applications. This work is documented in multiple recent publications [12, 13, 14, 15].

1.1.2 Stampede-based Distributed Systems Technologies

In this section we detail a number of activities related to the Stampede [16, 17, 18, 19] distributed programming infrastructure.

Media Broker Architecture. There was a time when the world of sensors and sensor-based processing was distinct from mainstream high-performance computing. It has become increasingly clear that these two worlds are inevitably merging as sensors and embedded systems become more ubiquitous in our environment. There is an interesting philosophical conflict in this merger. The sensor-driven DSP world knows the value of "throwing away" data as much as using sensitive data. The mainstream computing world, with its bag of tricks for consistency, check-pointing, etc., is very much concerned about keeping around data, expecting everything to be used! Given the confluence of sensor-driven and mainstream computing, there is a need to merge the technologies of the two worlds and perhaps discover unifying techniques for the merged sensor-based distributed computing fabric. In this merged world, interfacing and integrating with the physical environment will be a key property of high-performance distributed computing.

We expect more and more mainstream applications of the future to take on a "control loop" flavor in the following sense: inputs from a variety of distributed sensors may be gathered, filtered, processed, correlated, and fused to facilitate higher levels of hypotheses and inferences (implying heavy duty processing) which may then be used to drive distributed actuators to affect the physical world in some application-specified manner. Examples of such applications include mobile robots, smart vehicles, aware environments, disaster response, surveillance, and environmental monitoring.

We are building a *Media Broker* to provide adaptive naming, resource discovery and data fusion. The media broker (see Figure 1) acts as a clearinghouse for sensors and actuators, for data naming and efficient

resource discovery, and for specifying application specific behaviors such as sensor fusion, and is being built on top of the D-Stampede [17] system. We feel that naming and resource discovery should be elevated to the level of programming. Data fusion is a natural way to realize virtual sensors. For example, one can think of a virtual sensor that fuses corresponding inputs from two or more real sensors (such as audio and video). In this sense, data fusion is closely related to the naming problem: naming and resource discovery are common issues to be dealt with for both real and virtual sensors. Identification of the problems in constructing complex control loop applications and Initial ideas on addressing them have been reported in [20]. Two undergraduates Martin Modahl and Ilya Bagrak have been involved in the implementation of the Media Broker architecture.

Data Fusion. Technological innovation will, in the near future, deliver wireless, mobile sensor "motes" with the computation and communication capabilities of current handhelds. Thus, advanced multimedia applications that use sophisticated computing infrastructures today may become viable for deployment in future wireless ad hoc sensor networks. Simple in-network data aggregation techniques for sensor networks have been the focus of several recent research efforts. In this research, we extend these techniques to future sensor networks and ask two related questions: (a) what is the appropriate set of data aggregation (or fusion) techniques, and (b) how do we optimally assign aggregation roles to the nodes of a sensor network. We have developed an architectural framework, *DFuse*, for answering these two questions. It consists of a data fusion API and a distributed algorithm for energy-aware role assignment. The fusion API enables an application to be specified as a coarse-grained dataflow graph. It also eases application development and deployment. The role assignment algorithm maps the graph onto the physical network based upon the network topology and environmental conditions, periodically adapting the mapping in case of changes using role migration. Extensive measurements on an iPAQ handheld "farm" show that DFuse API calls incur low overhead. Preliminary results from this research are reported in [21].

Migration of Computational Abstractions. A key technology that is needed to support the DFuse architecture is on-demand migration of computational abstractions to match role assignment. We have designed and implemented facilities in Stampede for dynamically packaging and migrating Stampede channels without disrupting application progress.

Dead Timestamp Identification in Stampede. We are continuing joint work [22] uith with Dr. Kath Knobe of HP CRL, reported in last year's report, for identifying and eliminating irrelevant items in a Stampede computation. The measurement infrastructure coupled with the post-mortem analysis technique we have developed in Stampede allows us to ask several "what if" questions to validate or eliminate possible design directions for optimizing the performance of the Stampede runtime system.

Towards Aspect-Oriented Programming Support for Cluster Computing. This is continuation of an effort to automate certain aspects of distributed programming technologies, in particular the exploration of parallelization strategies and the plumbing among the computational entities. We are extending STAGES to provide these functionalities for cluster computations to a fully distributed framework with dynamic join/leave capabilities. We are also using this as a starting point for providing higher level debugging facilities in Stampede.

Stampede.NET. An effort started recently extends the Stampede programming model to wide-area networks using the Microsoft .NET infrastructure. The .NET substrate allows us to export Stampede capabilities to heterogeneous clients across the Internet as web services. The web services paradigm is predominantly used in the B2B domain for XML-based document exchange. Using this paradigm for Stampede-style

distributed stream processing opens new opportunities for stress testing the transport layer and evolving new protocols. .NET has mechanisms for supporting language heterogeneity and garbage collection. We are exploring ways to exploit these features for supporting the runtime model of Stampede.

1.1.3 Mechanisms for Embedded Architectures

Software Caching for Embedded Sensor Processing. This is continuation of work reported in the 2001-02 annual report to reconcile programmability with cost for embedded processing associated with sensors in a distributed sensing infrastructure.

Using our initial implementations of a SoftCache system for both Sparc and ARM/XScale systems [23, 24], optimization efforts are underway to reduce the penalties incurred by the SoftCache. Part of this work is attempting to characterize and predict moving from steady state to steady state without emulation of block invalidates ([26] contains some aspects of this work in which we have involved a female undergraduate Naila Farooqui).

Due to the complexity of debugging binary rewriting and translation systems with no *a priori* source code knowledge, the ARM/XScale version which uses a client-server methodology has been incorporated into SimpleScalar 4/ARM and gdb. SimpleScalar has been modified to support a large body of missing kernel functions (sockets, mmap, etc.), while gdb has been modified such that it can directly read and manipulate the server data structures to facilitate debugging. In this manner, translated addresses and code can be directly mapped back to the original program, and gdb sessions can be synchronized to understand the point of failure. Two undergraduates (Christopher J. Parnin and David Raeman) have been at the center of getting this work done.

To address the energy-delay issues that the SoftCache model forces on target devices, work has begun on addressing the different affected dimensions. After a review of methodology, we observed that the Energy Delay metric proposed by Gonzalez and Horowitz has been applied inconsistently. After addressing this issue by publishing a study of the problems [30], we have performed a detailed analysis of the energy-delay impact when replacing local memory with remote network accesses. Our results suggest that using the network as a remote memory can be more energy efficient than local memory under a wide range of conditions [25]. Additional analysis has been applied to the reorganization of the on-die structures, considering application performance and energy consumption of the processor overall [26].

Work is also moving forward on fully instrumenting an XScale platform (the ADI BRH from ADI Engineering, Inc.) to perform live power measurements while allowing for dynamic voltage- and frequency-scaling. Once completed, this instrumentation will allow for hardware verification a new analytical model. This modeling will account for individual power contributions to the overall system of DRAM, network, CPU, etc. This will permit correlation of simulated SoftCache power savings and network energy efficiency to typical hardware.

1.1.4 Sensor Technologies

Sensor Lab. We have provisioned a Sensor Lab within the College of Computing to support our research into flexible software infrastructure for pervasive, heterogeneous, sensor networks. Our lab includes modest quantities of a wide variety of devices, both wired and wireless, from simple commercially available (COTS) components to complex research prototypes.

In last year's report, we described in detail five specific sensor technologies we planned to deploy including an IR tag-based location system, passive temperature sensors, ultrasonic sensors, Berkeley Motes, and HP SmartBadge systems. All have been acquired and all but the ultrasonic sensors and Berkeley Motes have been deployed. In addition we have made extensive use of cameras, microphones and speakers and will be acquiring some RFID equipment and proximity cards for use in the Future Library project.

We have continued outfitting the Sensor Lab within the College of Computing and had a group of students develop a demo application that exercised the deployed sensor infrastructure. The Systems Studio and Systems Lab, along with other portions of the second floor of the College of Computing, have been outfitted with a commercial IR/RF tag-based tracking system. This system provides room-granularity tracking information on participants caring a tracking badge. Low-level sensor data is processed by a proprietary server which then emits tracking data in XML format to our Stampede infrastructure where it is published for applications. The Sensor Lab also contains a variety of microphones and cameras, with varying capabilities, including a pan-tilt-zoom camera. We have contact sensors on doors and passive temperature sensors. We have also been working with iPAQs and prototypes of the HP SmartBadge, which is a portable, wireless single-board computer with attached sensors that runs Linux. Finally we have started working with Berkeley Motes which provide a rich range of capabilities.

Our demo application involves tracking a target individual carrying an IR badge. This provides coarse location information (e.g. in the hallway, in the lab). we have sensors inside and outside doorways to detect entry/exit events. This data is further reinforced (via sensor fusion using the Fusion Channel API in Stampede) by door contact sensor readings. Once a tracked target is confirmed as entering the lab, a pan-tilt-zoom camera orients and takes a snapshot of the individual entering the room. This is stored for future reference. To track location within the lab, we have deployed simple login monitors on the various machines. Keystroke signatures are used to confirm that an individual logged into an account is the actual target. Using a simple spatial model of the sensored labs and spaces, appropriate cameras are identified to provide surveillance of the tracked target. This demo is mostly complete and exercises the infrastructure while providing a framework for further research. For example, part of the world model includes a sensor/actuator device registry. We have explored simple XML encodings of device characteristics for discovery, all within the Stampede framework. We look forward to augmenting this demo in novel ways with the Berkeley Motes. This demo used the HP SmartBadge as a computational node (although we didn't employ the onboard sensors) running Stampede.

The wireless temperature sensors and HP SmartBadge platform were both used by students in Professor Ramachandran's Pervasive Computing seminar (8803E) during Spring semester of 2003. We hope to see make the Sensor Lab more widely available to classes and students in the future.

We have done some preliminary evaluation of the Berkeley Motes but await delivery of several dozen new Motes and supporting infrastructure. Expected delivery is mid-summer 2003 and we hope to gear up quickly, training several undergraduate and graduate students in the use of the Motes and the TinyOS programming environment. Due to the flexibility of the Motes, many projects are possible. Early on we hope to develop a minimal Stampede protocol stack to run on the Motes, allowing them to integrate easily into existing applications and other research efforts.

Two other efforts are related to the Sensor Lab. First, to facilitate ongoing interaction with Professor Ron Arkin and the robotics group here at Georgia Tech, we plan on purchasing and deploying robots currently used by Ron Arkin's Mission Lab system. Robots are effectively autonomous, mobile sensors and provide a good application of the Stampede infrastructure. Having our own dedicated robotics hardware will allow us to progress more quickly with our efforts in this application domain. Second, in joint work with Professor Ramesh Jain, we will be deploying Stampede as part of a campus-wide Georgia Tech Event Web prototype. This system facilitates development and access to web-mediated events. In the Georgia Tech prototype, the system will be used to access talks, seminars, and conferences held on campus. This work will involve outfitting several spaces on campus with teleconferencing capabilities, effectively extending the Stampede-managed Sensor Lab to a campus-wide presence.

Software Infrastructure. Last year's report outlined a variety of infrastructure technologies that were planned or under development including basic sensor support, fusion channels, time synchronization, power

management, and the Media Broker. We have made progress on several fronts and some new research directions have crystallized in the intervening year.

We have demonstrated our ability to sample and control sensors and actuators within the Stampede framework with a variety of applications and demos. We have integrated the Fusion Channel abstraction into Stampede. We have undertaken a power management study and developed algorithms and infrastructure for dynamic power monitoring and management by task placement and migration (DFuse). Transparent task migration in Stampede required the development of a channel migration mechanism. This mechanism is useful independently and can be used to implement load-balancing and traffic management.

Efforts on the Media Broker are ongoing. A preliminary implementation has been used to support and structure the Family Intercom application in the Aware Home. This application allows the description and discovery of microphones and speakers throughout the Aware Home. These are often accessed in a combined form as a virtual sensor-actuator pair (ICombo). Information about the placement, attributes and status of devices is maintained in an LDAP repository. The LDAP repository also functions as a name server, allowing abstract named access to Stampede internal entities (e.g. channel descriptors, etc.). This information is augmented by a location tracking system in the Aware Home, accessed via a Java interface through the Context Toolkit. The Family Intercom supports tracking of listeners and speakers and activation of nearby audio resources. Ultimately, users will be able to move freely throughout the home with automated "switch-over" of audio devices as the user enters and leaves various rooms.

Development on the Media Broker proceeds on several fronts. We are adding mechanisms for stream publishers (sensors, producers) to specify the various data formats they can produce using a lattice-based type system. Data types are related by possible stream transformations in these lattices (for example, down-sampling, rate reduction, cropping, clipping, compression, etc.). Stream publishers also export a command set for changing from one format to another. Internally, the Media Broker is able to implement further transformations that are part of a library of well-known transformation, or via transforms that are provided by clients and dynamically linked on demand. The lattice structure helps the Media Broker determine a "greatest common format" when conflicting requests are presented. We are also exploring additional description and discovery techniques and are considering the use of DSML (Directory Service Markup Language) for representing LDAP data.

We have also undertaken a refactoring and redesign of Stampede to provide flexibility and performance enhancements and to add new features. This work involves careful object-oriented design and modeling and is proceeding with attention to current best software development practices. This effort will yield an improved and well-documented code base to support future enhancements. This work proceeds synergistically with an ongoing Stampede/.NET implementation in which a C# version of Stampede is being implemented within the .NET framework and exposed via web services. We have begun focusing initially on a redesign of the CLF communication substrate to modularly support a variety of transport mechanisms (UDP, reliable UDP, TCP, HTTP, etc.). We are redesigning this layer to provide a secure and flexible deployment/startup mechanism using SSH. In addition we are designing a dynamic, flexible group membership manager with basic fault tolerance. This will allow an improved, integrated implementation of the D-Stampede functionality (currently implemented via TCP proxying) as well as cluster-to-cluster interactions. The redesign will also support heterogeneous address spaces, eliminating the current requirement that all Stampede cluster participants host copies of a single statically-linked address space.

1.1.5 Distributed Ubiquitous Displays.

The aim of this project is solving some of the technical challenges in realizing a part of Marc Weiser's ubiquitous computing vision. The specific aspect of the vision being investigated is the ability to "whistle up" a display whenever and wherever. The vision is clearly beyond realization through a one-year project. The pieces of technology that were promised include: (a) techniques for a frontal multi-projection system

that automatically accounts for occlusion; (b) implementing this technique on the Stampede distributed system framework; and (b) mechanisms for data fusion in Stampede to allow multiple video streams to be combined for display purposes.

We made significant progress in increasing and assessing the usability of projector-camera technology. The original version of the system for occluder light suppression (turning off pixels that fall on the user) and shadow elimination (using alternate light paths to compensate for shadows) was based on a nonlinear feedback law that blended the outputs of multiple overlapping projectors to achieve a stable, occlusion-free display. While this process produced a working prototype, it required a polling loop in which each projector was tested in sequence to determine its effect on the image. This resulted in slow performance. We have since a developed a much simpler version of the system based on a switching controller. The key innovation is to assign all of the light for a given pixel to one of the unoccluded projectors. This eliminates the need for complex per pixel gain calculations and results in much faster performance. The current system operates around 12-15 Hz, a factor of 3-4 improvement over the original system. A paper describing the new method was presented this past June at the IEEE Conf. on Computer Vision and Pattern Recognition [27].

A second basic issue is to assess the usability of Virtual Rear Projection technology. There are a gamut of techniques which could be employed to achieve stable occlusion free displays. On one extreme is the switching approach described above which relies upon real-time visual feedback and complex algorithms. At the opposite extreme is what we call passive VRP, in which two overlapping aligned projectors display the same image at half power. In this setup each occluder casts two soft shadows on the display surface. The passive approach lies "between" conventional single projector setups which create hard shadows, and an active method which can cancel shadows completely. From a deployment perspective this gamut of approaches represents a gamut of implementation costs and complexities. We believe it is important to assess the relative value of these alternatives.

This past Spring Jay Summet, a Ph.D. student, conducted the first user-study of VRP technologies. Since the active VRP is still under development, Jay focused on establishing a baseline by comparing passive VRP and two single-projector display solutions along with true rear projection. His study assessed usability as well as user preferences through a variety of selection tasks. The study was implemented using a Smart Board rear projection display with a touch sensitive screen. For VRP technologies, light was projected onto the Smart Board surface. For the rear projection component the Smart Board was used directly. The most significant finding from the study was a strong user preference for VRP technology over other single projector solutions. This suggests that the added complexity and cost of using a second projector can be justified in terms of the user-experience that it provides. The study and its results are described in a paper which has been submitted to the First Intl. Workshop on Projector-Camera Systems (PROCAMS 03), which will be held this October in Nice, France [29].

One limitation of the study was the need to use the touch screen surface of the Smart Board for all interactions. Since the Smart Board surface is designed for back projection use, it is a relatively poor surface for front projection. This placed all of the front projection approaches at a disadvantage with regard to the image quality. We plan to conduct a more involved study this year which will include the switching VRP technology. This study will use a new pen-based capture product which is being custom-built for us by Smart Technologies. This system will make it possible to capture from ordinary projection surfaces.

A journal paper describing an overview of the Stampede system and an evaluation of its performance was recently accepted for publication in the IEEE Transactions on Parallel and Distributed Systems special issue on middleware [16]. This evaluation, completed in Fall 2002, assessed the performance of two diverse applications on our Stampede implementation using the Beowulf cluster in CERCS. It demonstrates the ability to achieve significant speedups on streaming media applications under Stampede.

An implementation of the core projector-camera system under the Stampede system is currently underway by Xiang Song, a first year Ph.D. student. Xiang is basing his port on a Windows implementation of Stampede which uses MPI as the message-passing layer. The Windows version is important in order to be

able to use the most recent graphics cards in the system, and to avoid extensive modification of the code. We have made significant improvements to the Windows version of Stampede in response to the needs of this application. This portion of the project will be completed by the end of the Summer. A preliminary demo of the Windows Stampede system using .NET technology will be presented at the Microsoft Research Summit in July.

1.1.6 Performance Study of Stampede Runtime System.

Emerging application domains such as interactive vision, animation, and multimedia collaboration display dynamic scalable parallelism, and high computational requirements, making them good candidates for executing on parallel architectures such as SMPs or clusters of SMPs. Apart from their main algorithmic components, these applications need specialized support mechanisms that enable plumbing different modules together, cross module data transfer, automatic buffer management, synchronization and so on. Such support mechanisms are usually part of the runtime system, and we have developed techniques to quantify their performance. The runtime for our evaluation is *Stampede*, a cluster programming system that is designed to meet the requirements of such applications. We have developed a cycle accurate timing infrastructure that helps tease out the time spent by an application in different layers of software, viz., the main algorithmic component, the support mechanisms, and the raw messaging. We conducted our experiments with two representative applications on two flavors of Unix, viz., Solaris and Linux. There are several interesting insights coming from this study [28], published in the *International Symposium on Performance Analysis of Systems and Software, 2003*. First, memory allocation does not take up a significant amount of the execution time despite the interactive and dynamic nature of the application domain. Second, the Stampede runtime adds a minimal overhead over raw messaging for structuring such applications. Third, the results suggest that the thread scheduler on Linux may be more responsive than the one on Solaris; we quantify the effect of the scheduler responsiveness in terms of difference in *blocking time* for threads and time spent in synchronization in the messaging layer. Fourth, the messaging layer spends quite a bit of time in synchronization operations. Perhaps the most interesting result of this study is that general-purpose operating systems such as Linux and Solaris are quite adequate to meet the requirements of emerging dynamic interactive stream-oriented applications.

1.1.7 SOLAR: Semantic-Oriented Low-power Architecture

Semantics-Aware Multilateral Partitioning. This work [30, 31] explores and implements a novel low power memory technique for a microprocessor through an in-depth characterization study for memory reference behaviors and locality to reduce energy consumption in data TLBs and caches. We address the memory energy issues by employing a streamlined partitioning technique at architecture level that effectively reduces energy consumption in memory subsystem without compromising performance. It is achieved by decoupling the d-TLB lookups and the data cache accesses, based on the semantic regions defined by the convention of programming languages on a given architecture, into discrete reference substreams ¿ stack, global static, and heap. It is observed that data memory accesses from each semantic region form a semantic band. The heap band is wider and denser than the stack and global static bands indicating the substantially higher number of unique heap pages accesses. Also it is found that only a very small number of stack virtual pages is needed compared to the requirement for global static and heap pages. While mapping all these references to the same structure, unnecessary conflict misses among different semantic regions disturb both performance and power consumption. This motivates us to redesign the memory subsystem by employing a semantic-aware multilateral (SAM) partitioning, which customizes a discrete memory structure for each semantic region to eliminate conflict misses. The SAM memory architecture exploits the locality and characteristics demonstrated in each semantic region by (1) Semantic-Aware TLBs (SAT), reorganizing the first

level TLB structure into two small structures – stack and global static micro-TLBs, while leaving the second level for all data addresses and (2) Semantic-Aware Caches (SAC), by decoupling the first level cache into three cachelets. A Data Address Router (DAR), containing semantic boundary addresses in special control registers, was implemented to route each d-TLB lookup to their respective SAT for address translation. Instead of searching the entire fully associative TLB entries, energy can be saved by eliminating conflict misses and filtering out the lookups by stack and global static regions with smaller content-addressable memory structures as a major portion of the memory access distribution being skewed toward the much smaller stack-TLB and global-TLB. Similar to SAT, SAC also saves substantial energy by redirecting the majority of memory references to stack-Cachelet and global-Cachelet. In addition, multi-porting the smaller yet frequently accessed SAT and SAC requires less die area, as well as reducing the access latency. Our experimental results show that an average of 35% energy can be reduced in the d-TLB and the data cache. Furthermore, an average of 45% energy can be saved by selectively multi-porting the Semantic-aware d-TLBs and data caches against their monolithic counterparts.

Minimizing Energy using Multiple Supply and Threshold Voltages. This work [32] proposed an optimum methodology for assigning multiple supply and threshold voltages to different modules in a CMOS circuit such that the overall static and dynamic energy consumption is minimized for a given delay constraint. The theoretical conditions for minimum energy were identified mathematically by applying the Lagrange Multiplier algorithm using the overall delay of the modules on the critical path as a constraint followed by our proposed gradient search algorithm that iteratively converges to optimal supply and threshold voltages that satisfy the optimum energy. We also presented a clustering algorithm which clusters these optimum values into a limited number of supply and threshold voltages in a practical design scenario. The method can be applied to circuit modules of any kind, given the delay and energy parameters for the modules. We applied our algorithms to using 0.25ţ process technology and observed up to 48% energy savings in ISCAS'85 benchmark and a 16-bit Wallace Tree Multiplier.

1.1.8 Efficient Network Emulation

This project is concerned with realizing novel, next-generation techniques that will enable rapid emulation/simulation of large-scale networks for use in on-line simulation-based network management methods, and to facilitate distributed application development, analysis and testing. The ultimate objective is to demonstrate the use of this technology in enabling new capabilities in defense communication applications.

We are developing key technologies to achieve our objectives:

- A novel parallel software backplane that enables model reuse from different network simulation and emulation tools
- New parallel discrete event simulation techniques to enable execution of massive simulations of large-scale network models, on multiple platforms including supercomputers
- New techniques exploiting direct execution for accurate and transparent capture and control of live application network activity
- Integrated demonstrations, with multiple NMS research groups, actively identifying and showing military relevance.

Recent accomplishments [33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44] include:

Hierarchical Time Synchronization Algorithm. Designed and implemented a novel, hierarchical time synchronization algorithm that can scale to more than a thousand processors. Used this algorithm in our supercomputer simulations to achieve the fastest simulations to date (106 million packet transmissions simulated in one wallclock second). The implementation is included in our backplane software, the Large-scale Network Visualization Tool. Developed a Java-based scalable network visualization tool, NetAnim, for rapidly visualizing the topology and animating the traffic in very large network configurations, with novel algorithms for fast layout. Unlike existing tools, our NetAnim tool scales to hundreds of thousands of nodes, links and packets. Used the tool for visualizing million-node military enterprise networks.

GTNetS – New Network Simulator. New parallel network simulator which contains more detailed models, is built for interoperability from the outset, is demonstrated to be highly scalable, and implements memory-optimized models for large-scale configurations.

Hybrid Packet-Fluid Simulations. Developed a hybrid approach to combine high fidelity of packet-level models with simulation speed of fluid simulations. Achieved a computational speedup of over twenty in some cases, with little loss of accuracy.

Multicast Nix Vectors. Developed a novel representation technique that greatly reduces memory requirements for routing in packet-level multicast simulations. Our representation enables the simulation of large networks which were not possible to simulate before due to very high memory overhead of traditional representation techniques.

1.1.9 Head and Gaze Tracking for Effective Interaction

Head pose and eye gaze information are very valuable cues in face-to-face interactions between people. Such information is also important for computer systems that a person intends to interact with. The awareness of eye gaze provides context to the computer system that the user is looking at it and therefore supports effective interaction with the user.

Real-time Vision-based Eye Tracking System. We have developed [45] a real-time vision-based eye tracking system to robustly track a user's eyes and head movements. Our system utilizes robust eye tracking data from multiple cameras to estimate 3D head orientation via triangulation. Multiple cameras support a larger tracking volume than is possible with an individual sensor, which is valuable in an attentive environment. Each individual eye tracker exploits the red-eye effect to track eyes robustly using an infrared lighting source. Although our system relies on infrared light, it can still track reliably in environments saturated with infrared light, such as a residential living room with much sunlight. For the latter, we have developed a new Fischer Discriminant for eye images from IR cameras.

Speaker and Movement Tracking with Audio/Video Fusion. In addition, we have also studied [46] several approaches for fusion of audio and video streams to allow for real-time tracking of moving and speaking people. This requires building off existing stampede architecture and its related Media Broker to allow for efficient streaming and processing of these rich streams. We have built several phased-array microphone systems and multiple-camera trackers in support of this effort.

1.1.10 High-Performance and Low-Power SOC and SOP Technologies

High-performance and low-power hardware substrates facilitate pervasive deployment of streaming media systems. To this end, we have been exploring logic synthesis and physical design for high performance,

low-power, and reliable SOC (system-on-chip) and SOP (system-on-package) technologies. Three recent projects are described below.

Thermal-driven Circuit Partitioning and Floorplanning with Power Optimization. In this work [47], we present methodology to distribute the temperature of gates evenly on a chip while simultaneously reducing the power consumption by using newly designed partitioning and floorplanning algorithms. This new partitioning algorithm is designed to partition blocks with well-balanced temperatures by altering the FM algorithm to include thermal constraints. Then, the suggested floorplanning algorithm can assign specific geometric locations to the blocks to refine the quality of the thermal distribution and to reduce power consumption. The combination of these two new algorithms, called TPO, is compared with the results of a conventional design procedure. As a result, power is reduced by up to 19on average and a well-distributed thermal condition is achieved.

Simultaneous Delay and Power Optimization for Multi-level Partitioning and Floorplanning with Retiming. Delay minimization and power minimization are two important objectives in the design of the high-performance, portable, and wireless computing and communication systems. Retiming is a very effective way for delay optimization for sequential circuits. In this work [48], we propose a unified framework for multi-level partitioning and floorplanning with retiming, targeting simultaneous delay and power optimization. We first discuss the importance of retiming delay and visible power as opposed to the conventional static delay and total power for sequential circuits. Then we propose GEO-PD algorithm for simultaneous delay and power optimization and provide smooth cutsize, wirelength, power and delay tradeoff. In GEO-PD, we use retiming based timing analysis and visible power analysis to identify timing and power critical nets and assign proper weights to them to guide the multi-level optimization process. In general, timing and power analysis are done at the original netlist while a recursive multi-level approach performs partitioning and floorplanning on the sub-netlist as well as its coarsened representations. We show an effective way to translate the timing and power analysis results from the original netlist to a coarsened sub-netlist for effective multi-level delay and power optimization. To the best of our knowledge, this is the first work addressing simultaneous delay and power optimization in multi-level partitioning and floorplanning.

Physical Layout Automation for System-On-Packages. System-On-Package (SOP) paradigm proposes a unified chip-plus-package view of the design process, where heterogeneous system components such as digital ICs, analog/RF ICs, memory, optical interconnects, MEMS, and passive elements (RLC) are all packaged into a single high speed/density multi-layer SOP substrate. We propose a new chip/package co-design methodology [49] for physical layout under the new SOP paradigm. This new methodology enables the physical layout design and analysis across all levels of the SOP design implementation, bridging gaps between IC design, package design, and package analysis to efficiently address timing closure and signal integrity issues for high-speed designs. In order to accomplish a rigorous performance and signal integrity optimization, efficient static timing analysis (STA), signal integrity analysis (SIA), and thermal and power analysis (TPA) tools are fully integrated into our co-design flow. Our unified wire-centric physical layout toolset that includes on-chip/package wire generation, on-chip/package floorplanning, and on-chip/package wire synthesis provides wire solutions for all levels of the design hierarchy-including cell, block, and chip level for pure digital and mixed signal environment. In addition, on-chip hard/soft IP (Intellectual Property) integration is supported in our co-design flow for shorter design times through design reuse. To the best of our knowledge, this work is the first to address the chip/package co-design issues in System-On-Package (SOP) physical layout.

1.2 Major Findings

We have had a significant number of research results in the scope of the RI project that span distributed systems, user interaction, cluster computing, and wide area scalability in application domains such as telepresence and e-commerce. These results have often been in partnership with our industrial collaborators such as IBM, Intel, HP, Microsoft, and Wind River Systems. The references in the publications portion of this annual report give pointers to some of these results.

1.3 Training and Development

1.3.1 Undergraduate Research Participation

We continue to attract bright and interested undergraduates to RI-related research projects. Undergraduate participation in research within the College is facilitated by the excellent UROC program (www.cc.gatech.edu/program/uroc), coordinated by Amy Bruckmann. A variety of institute-wide programs are also available (www.undergraduateresearch.gatech.edu) including a special fund sponsored by the president of Georgia Tech and several NSF-sponsored projects (URIP, SURF, etc.). In addition, we sponsor the Systems Hackfest group each semester that includes 5-10 undergraduates participating in research-related projects for fun or course credit.

1.4 Outreach

We have held several open-house events in 2002 to acquaint researchers from within and outside the College of Computing to the high-end visualization and computation facilities that are available through the RI award. The Systems Studio has become a center for demo and poster events and is used frequently to introduce and showcase our various efforts to visitors.

In Fall 2001, the Georgia Tech Center for Experimental Research in Computer Systems (CERCS) opened its doors. CERCS brings together researchers from Georgia Tech's College of Computing and School of Electrical and Computer Engineering who share a common focus on the design and evaluation of computer and software systems through experimental methods. CERCS research focuses on complex systems, including their hardware, communications and system-level software, and applications. By emphasizing the experimental method, we promote the creation of knowledge through the design, implementation, and measurement of potentially large-scale prototype systems. Constituting one of the largest experimental systems programs in the U.S., CERCS has a mission to:

- promote experimental research in Computer and Software Systems
- produce high quality students trained in the experimental method of systems research and development
- foster high impact and multi-disciplinary research efforts using shared personnel and facilities, and
- support researchers and educators at Georgia Tech and its affiliated institutions.

We received an NSF IUCRC award toward the formation of this center. In Fall 2002, we held the formal opening of the center and dove-tailed it to the NSF IUCRC workshop on experimental research in computer systems.

1.4.1 Advisory Board

We held a valuable research advisory board for the RI award in October 2002. The board members attending included: Roger Haskin (IBM Almaden), Kath Knobe (formerly at Compaq CRL), Jim Rowson (HP labs), Rick Schlichting (AT & T), Willy Zwaenepoel (Rice), Margaret Martonosi (Princeton), and Mary Vernon (UW-Madison). This meeting provided valuable external feedback and focus to our efforts. We look forward to the next meeting of this group, tentatively scheduled for Spring of 2004.

2 Publications and Products

2.1 Publications

See the references at the end of this document.

2.2 Web Site

Please visit the project web site at www.cc.gatech.edu/~rama/nsf-ri/

3 Contributions

The activities we are currently undertaking have resulted in a significant number of publications and software artifacts. These are listed in the references at the end of this report.

3.1 Human Resources

We have roughly 40 graduate students, 15 undergraduate students, and 7 research scientists working in areas related to this project. Three research scientists are supported (part-time) directly from grant funds. Approximately 30% of these individuals come from under-represented groups.

3.2 Research and Education

The research artifacts from the project are finding their way into graduate courses. Professor Ramachandran taught a special topics course entitled, "Pervasive computing with distributed sensors," (URL: `www.cc.gatech.edu/classes/AY2002/cs8803e_spring`). Some of the students in the course used D-Stampede as an implementation vehicle for the course project.

Courses using RI equipment and updated in response to RI-enabled research: CS 6210, CS 6235, CS 6230, CS 3210, CS 4210, CS 4220.

3.3 Laboratory and Testbed Development

We have outfitted a laboratory (the Systems Studio) with a variety of resources described in previous annual reports. We can report ongoing heavy use of these resources for a variety of grant-related projects. This "media space" is used frequently for interactive meetings and presentations and has been outfitted as an Access Grid node and is used regularly for this purpose by individuals from throughout the Tech campus.

This year we have purchased a variety of additional sensor components so that we may use the Systems Studio as a "sensor lab" as well. We are experimenting with both research (HP Badge4, Berkeley Motes) and commercial equipment (VersusTech location sensors, wireless temperature sensors).

In addition, we are increasing the integration of resources in the Systems Studio with similar resources in the Aware Home.

3.4 Budget, Equipment Acquisition, and Personnel Support

Of the $300,018 increment received this year from NSF, we report expenses of $285,643.82. The bulk of this amount ($225,000) was spent on the purchase of a large HP Itanium cluster which was described in detail in last year's report. The cluster is now operational and in daily use. A second expense of approximately $34,000 purchased a high-speed Cisco switch which provides access to the Itanium cluster. The remaining expense (approximately $27,000) was a combination of smaller expenses for small systems, maintenance, and personnel.

The current year's allocation of matching funds comes from two sources internally and totals $174,769.02 (College Matching Fund) and $86,000 (Liotta Matching Fund). College matching funds have been encumbered for equipment maintenance on RI purchases and for personnel (research scientist) support. Approximately $60,000 of the Liotta funds were applied to the Itanium cluster purchase (a large, multi-grant purchase). We purchased approximately $12,000 of Berkeley Motes and development systems from Crossbow Technologies also from the Liotta funds. We purchased a large tertiary storage system from Sun Microsystems to support backup of RI-funded storage media with an additional unspent $60,000 of Liotta matching funds. Finally, an additional $34,000 of unspent Liotta funds were allocated to small equipment purchases and maintenance.

Three research scientists are currently partially supported by the RI funds: Neil Bright, Phillip Hutto and Matthew Wolenetz. We project these three individuals will remain funded by the grant (primarily from College Matching Fund) for the duration of the grant.

4 Special Requirements

4.1 Plans for the Coming Year

Projects such as Stampede, ECho, and InfoSphere we have been exploring several interesting distributed systems technologies. Now we are a point where we can see the impact of these technologies in various application contexts. Such contexts include the Media Broker application described previously, a novel application under development called the TV-Watcher that monitors a large number of video and text streams for content correlation, and non-occluding ubiquitous interactive displays. In addition, using the sensor lab that we have outfitted, we plan on exploring sensor monitoring and fusion technologies to enable capabilities such as location awareness. As mentioned previously we will continue our efforts at integrating sensor infrastructure technologies in the Aware Home and the Systems Studio. We believe this effort will reveal interesting requirements due to the variety of technologies involved.

References

[1] Matt Wolf, Zhongtang Cai, Weiyun Huang, and Karsten Schwan. "Smart Pointers: Personalized Scientific Data Portals in Your Hand", *Supercomputing 2002*, ACM, November 2002.

[2] Qi He and Karsten Schwan. "Coordinating Application Adaptation with Network Transport", *Proc. of the 11th High Performance and Distributed Computing Conference (HPDC-11)*, July 2002.

[3] Patrick Widener, Greg Eisenhauer and Karsten Schwan. "Efficient XML-Based Communication for High Performance Computing", *High Performance Distributed Computing (HPDC-10)*, August 2001.

[4] Fabian Bustamante, Patrick Widener and Karsten Schwan. "Scalable Directory Services Using Proactivity", *Supercomputing 2002*, ACM, November 2002.

[5] Xiaotong Zhuang, Weidong Shi, Indrani Paul and Karsten Schwan. "Efficient Implementation of the DWCS Algorithm on High-Speed Programmable Network Processors", *IEEE Multimedia Networks and Systems (MMNS)*, October 2002.

[6] Raj Krishnamurty, Sudhakar Yalamanchili, Karsten Schwan and Rich West. "Scheduling Gigabit Packet Streams", *Proceedings of the 10th International Symposium on Hot Interconnects (HOT-I 2002)*, August 2002.

[7] Ada Gavrilovska, Kenneth Mackenzie, Karsten Schwan, and Austen McDonald. "Stream Handlers: Application-specific Message Services on Attached Network Processors", *Proceedings of the 10th International Symposium on Hot Interconnects (HOT-I 2002)*, August 2002.

[8] Jasmina Jancic, Christian Poellabauer, Karsten Schwan, Matthew Wolf, and Neil Bright. "Extensible Run-Time Resource Monitoring for Cluster Applications", *Proceedings of the International Conference on Computational Science (ICCS '02)*, Amsterdam, The Netherlands, April 2002.

[9] Dong Zhou, Karsten Schwan, and Santosh Pande. "Method Partitioning - Runtime Customization of Pervasive Programs without Design-time Application Knowledge", *International Conference on Distributed Computing Systems*, June 2003.

[10] Yuan Chen and Karsten Schwan. "Opportunistic Channels: Mobility-aware Event Delivery", *Middleware 2003*, June 2003.

[11] Greg Eisenhauer, Fabián Bustamante and Karsten Schwan. "Native Data Representation: An Efficient Wire Format for High-Performance Computing", *IEEE Transactions on Parallel and Distributed Systems*, to appear 2003.

[12] Christian Poellabauer, Hasan Abbasi, and Karsten Schwan. "Cooperative Run-time Management of Adaptive Applications and Distributed Resources", *Proceedings of the 10th ACM Multimedia Conference*, Juan-les-Pins, France, December 2002.

[13] Christian Poellabauer and Karsten Schwan. "Power-Aware Video Decoding using Real-Time Event Handlers", *Proceedings of the 5th International Workshop on Wireless Mobile Multimedia (WoWMoM)*, Atlanta, Georgia, September 2002.

[14] Christian Poellabauer and Karsten Schwan. "Kernel Support for the Event-based Cooperation of Distributed Resource Managers", *Proceedings of the 8th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2002)*, San Jose, California, September 2002.

[15] Fabian E. Bustamante, Christian Poellabauer, and Karsten Schwan. "AIMS: Robustness Through Sensible Introspection (Extended Abstract)", *Proceedings of the 10th ACM SIGOPS European Workshop*, Saint-Emilion, France, September 2002.

[16] U. Ramachandran, R. Nikhil, J. Rehg, Y. Angelov, A. Paul, S. Adhikari, K. Mackenzie, N. Harel and K. Knobe. "Stampede: A Cluster Programming Middleware for Interactive Stream-oriented Applications", To appear in *IEEE Transactions on Parallel and Distributed Systems*, Special Issue on Middleware, 2003.

[17] Sameer Adhikari, Arnab Paul, Bikash Agarwalla and Umakishore Ramachandran. "D-Stampede: Distributed Programming System for Ubiquitous Computing", Submitted to *IEEE Transactions on Parallel and Distributed Systems*, 2003.

[18] Arnab Paul, Sameer Adhikari and Umakishore Ramachandran. "An Implementation of a Cluster Runtime System for Streaming and Multimedia Applications", Submitted to *Software Practice & Experience*, 2003.

[19] A. Paul, N. Harel, S. Adhikari, B. Agarwalla, U. Ramachandran and K. Mackenzie. "Interaction between Stampede Runtime and Operating Systems", *IEEE International Symposium on Performance Analysis of Systems and Software*, March, 2003.

[20] Umakishore Ramachandran, Phillip Hutto, Bikash Agarwalla, and Matthew Wolenetz. "A System Architecture for Distributed Control Loop Applications", *Workshop on Future Trends in Distributed Computing Systems*, May, 2003.

[21] R. Kumar, M. Wolenetz, B. Agarwalla, J. Shin, P. Hutto, A. Paul and U. Ramachandran. "DFuse: A Framework for Distributed Data Fusion", *Proceedings of ACM SenSys '03*, Los Angeles, California, November 2003.

[22] Hasnain Mandviwala, Nissim Harel, Kathleen Knobe and Umakishore Ramachandran. "Distributed Garbage Collection for Timestamped Data", To be submitted to *IEEE Transactions on Parallel and Distributed Systems*, 2003.

[23] Chad Huneycutt, Joshua Fryman and Kenneth Mackenzie. "Software Caching using Dynamic Binary Rewriting for Embedded Devices", *International Conference on Parallel Processing 2002*, August, 2002.

[24] Joshua Fryman, Chad Huneycutt and Kenneth Mackenzie. "Investigating a SoftCache via Dynamic Rewriting", *Fourth Workshop on Feedback-Directed and Dynamic Optimization (FDDO)*, 2001.

[25] J. Fryman, C. Huneycutt, H. Lee, K. Mackenzie and D. Schimmel. "Energy Efficient Network Memory for Ubiquitous Devices", Technical Report GIT-CERCS-03-05, Center for Experimental Research in Computing Systems, Georgia Tech, Atlanta, Georgia, 2003.

[26] J. Fryman, H. Lee, C. Huneycutt, N. Farooqui, K. Mackenzie and D. Schimmel. "SoftCache: A Technique for Power and Area Reduction in Embedded Systems", Technical Report GIT-CERCS-03-06, Center for Experimental Research in Computing Systems, Georgia Tech, Atlanta, Georgia, 2003.

[27] T. J. Cham, J. M. Rehg, R. Sukthankar, and G. Sukthankar. "Shadow Elimination and Occluder Light Suppression for Multi-Projector Displays", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Madison, WI, June 2003.

[28] Arnab Paul, Nissim Harel, Sameer Adhikari, Bikash Agarwalla, Umakishore Ramachandran, and Ken Mackenzie. "Performance Study of a Cluster Runtime System for Dynamic Interactive Stream-oriented Applications", *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS-03)*, Austin, Texas, March 2003.

[29] J. Summet, M. Flagg, J. M. Rehg, G. D. Corso, and G. D. Abowd. "Increasing the Usability of Virtual Rear Projection Displays", Submitted to *First International Workshop on Projector-Camera Systems (PROCAMS 2003)*, Nice, France, 2003.

[30] Hsien-Hsin S. Lee, Joshua B. Fryman, A. Utku Diril, and Yuvraj S. Dhillon. "The Elusive Metric for Low-Power Architecture Research", In the *Workshop on Complexity-Effective Design (WCED-03)* in conjunction with the *30th International Symposium on Computer Architecture*, San Diego, California, June, 2003.

[31] Hsien-Hsin S. Lee and Chinnakrishnan S. Ballapuram. "Energy Efficient D-TLB and Data Cache using Semantic-Aware Multilateral Partitioning", To appear in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED'03)*, Seoul, Korea, August, 2003.

[32] Yuvraj Dhillon, Abdulkadir Diril, Abhijit Chatterjee, and Hsien-Hsin Lee. "An Algorithm for Achieving Minimum Energy Consumption in CMOS Circuits Using Multiple Supply and Threshold Voltages at the Module Level", To appear in *Proceedings of the International Conference on Computer-Aided Design (ICCAD-03)*, San Jose, California, November, 2003.

[33] Q. He, M. Ammar, G. Riley, H. Raj and R. Fujimoto. "Mapping Peer Behavior to Packet-level Details: A Framework for Packet-level Simulation of Peer-to-Peer Systems", *Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, October 2003.

[34] H. Wu and R. M. Fujimoto. "Time-Parallel Trace-Driven Simulation of CSMA/CD", *Workshop on Parallel and Distributed Simulation*, June 2003.

[35] T. McLean and R. M. Fujimoto. "Predictable Time Management for Real-Time Distributed Simulation", *Workshop on Parallel and Distributed Simulation*, June 2003.

[36] W. Shi, K. Perumalla and R. M. Fujimoto. "Power-aware State Dissemination in Mobile Distributed Virtual Environments", *Workshop on Parallel andDistributed Simulation*, June 2003.

[37] K. Perumalla, A. Park, R. M. Fujimoto and G. Riley. "Scalable RTI-Based Parallel Simulation of Networks", *Workshop on Parallel and Distributed Simulation*, June 2003.

[38] D. Xu, M. Ammar, G. Riley and R. M. Fujimoto. "Enabling Large-Scale Multicast Simulations by Reducing Memory Reqirements", *Workshop on Parallel and Distributed Simulation*, June 2003.

[39] S. Taylor, A. Bruzzone, R. M. Fujimoto, B. P. Gan, R. Paul and S. Strassburger. "Distributed Simulation and Industry: Potentials and Pitfalls", *Winter Simulation Conference*, December 2002.

[40] Q. He, M. Ammar, G. Riley and R. M. Fujimoto. "Exploiting the Predictability of TCP's Steady-state Behavior to Speed Up Network Simulation", *Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, October 2002.

[41] G. Riley, T. Jaafar and R. M. Fujimoto. "Integrated Fluid and Packet Network Simulations", *Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, October 2002.

[42] K. Perumalla and R. M. Fujimoto. "Using Reverse Circuit Execution for Efficient Parallel Simulation of Logic Ciruits", *SPIE 47th Annual Conference*, July 2002.

[43] S. Ferenci, R. Fujimoto, G. Riley, K. Perumalla and M. Ammar. "Updateable Simulations of Communication Networks", *Workshop on Parallel and Distributed Simulation*, May 2002 winner, best paper award.

[44] K. Perumalla, R. Fujimoto, T. McLean and G. Riley. "Experiences Applying Parallel and Interoperable Network Simulation Techniques in On-Line Simulations of Military Networks", *Workshop on Parallel and Distributed Simulation*, May 2002.

[45] Ravikrishna Ruddarraju, Antonio Haro, and Irfan Essa. "Fast Multiple Camera Head Pose Tracking", *Proceedings Vision Interface 2003*, Halifax, Canada. 2003.

[46] Ravikrishna Ruddarraju, Antonio Haro, and Irfan Essa. "Perceptual User Interfaces using Vision-based Eye Trackers", *Proceeding of International Conference on Multimodal and Perceptual User Interfaces*, October 2003, Vancouver, BC, Canada, 2003.

[47] Kyoung-Keun Lee, Edward J. Paradise, and Sung Kyu Lim. "Thermal-driven Circuit Partitioning and Floorplanning with Power Optimization", Technical Report GIT-CERCS-03-07, Center for Experimental Research in Computing Systems, Georgia Tech, Atlanta, Georgia, 2003.

[48] Mongkol Ekpanyapong, and Sung Kyu Lim. "Simultaneous Delay and Power Optimization for Multilevel Partitioning and Floorplanning with Retiming", Technical Report GIT-CERCS-03-08, Center for Experimental Research in Computing Systems, Georgia Tech, Atlanta, Georgia, 2003.

[49] Sung Kyu Lim. "Physical Layout Automation for System-On-Packages", Technical Report GIT-CERCS-03-09, Center for Experimental Research in Computing Systems, Georgia Tech, Atlanta, Georgia, 2003.

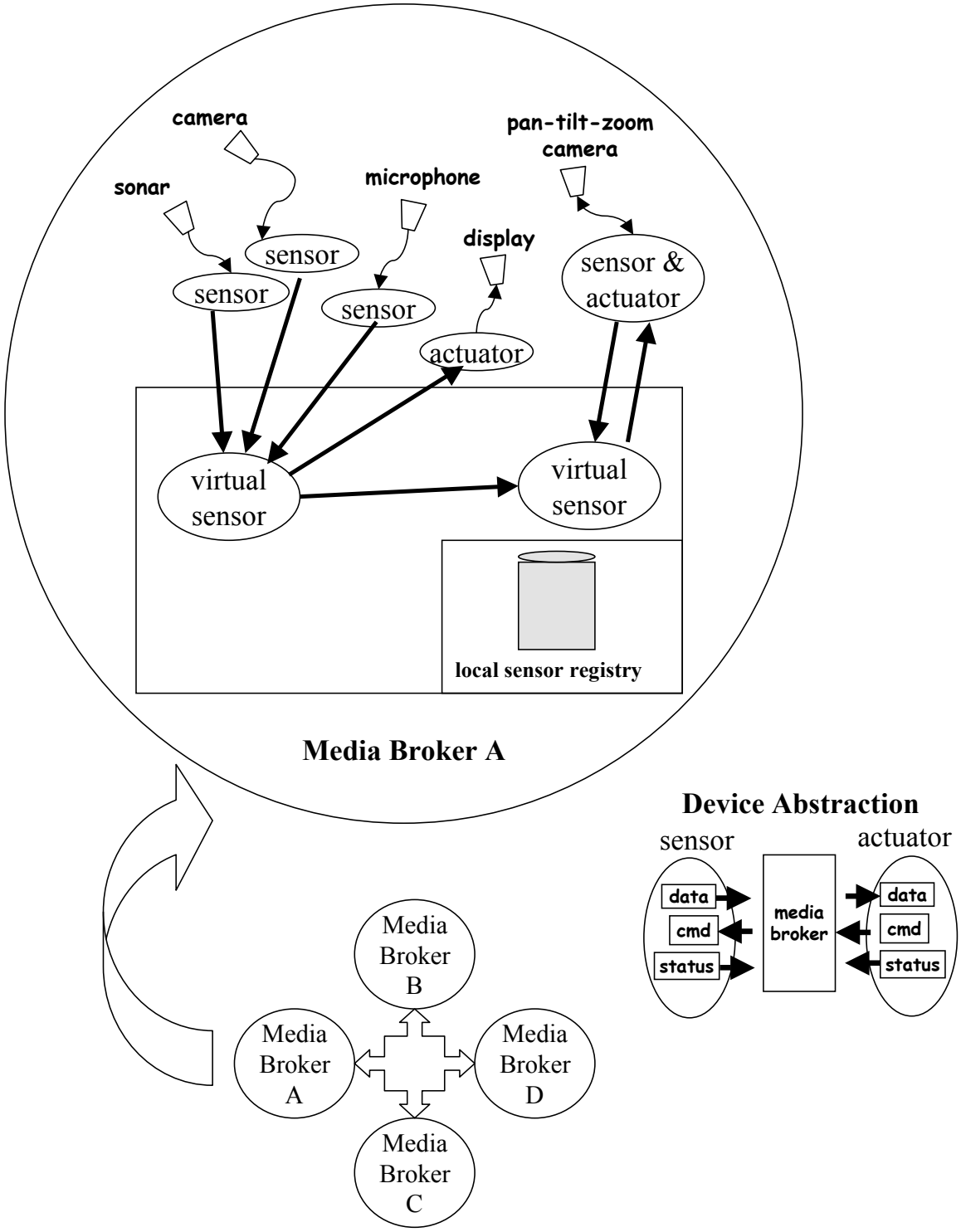

Figure 1: Media Broker Architecture