

# Research Statement

## Gregory D. Abowd

My entire research career has targetted the intersection between Software Engineering and Human-Computer Interaction (HCI). I am interested in pursuing challenging problems that involve the construction and human impact of computing technology. The vast majority of computing technologies are used by humans, so techniques to facilitate the design, implementation and evaluation of these systems is of obvious importance. Before arriving at Georgia Tech, my research style was classical with respect to both Software Engineering and HCI. There was no inherent problem with the method of my research. The problem I perceived was that I was inclined to apply classical techniques to classical problems. I was developing techniques to help people build the kinds of systems that they had been building for 10-15 years already. To heighten the chance that my research would have impact in either the HCI or Software Engineering research or practicing communities, I decided to shift my focus toward applications and computing systems that were not currently commonplace. My research would then serve the role of a leading indicator on computing practice and not a lagging indicator.

My research direction was influenced by the writings and work of the late Mark Weiser and his Ubiquitous Computing research project at Xerox PARC. Mark and his colleagues at PARC demonstrated Alan Kay's spirit of prediction by creating and using an environment enriched with new kinds of interaction devices, from handheld to wall-sized. They showed how a number of significant Computer Science research problems are unveiled when computation is spread throughout an environment and the use of the computation goes beyond traditional desktop computing tasks. In examining this seminal work at PARC, however, I noticed two problems:

- There was relatively little attention paid to understanding the long-term use of applications, with the notable exception of the Tivoli project's meeting capture.
- There was little effort to make the construction of these ubiquitous computing applications themselves easier so that more researchers could develop them.

Addressing these two problems best characterizes my approach to research in ubiquitous computing. With an applications focus, I have been developing techniques to build robust and evolvable ubiquitous computing systems and then observing the long-term use of these systems as they are absorbed into everyday activities. The general lessons learned over these past five years have been recently published in two major forums. First, a discussion of software engineering research issues for ubiquitous computing was presented at the International Conference on Software Engineering in May 1999 [C.23]. Second, jointly with Mynatt, I have surveyed much of the progress in ubiquitous computing over the past decade and highlighted four major application themes for HCI researchers as well as additional challenges to understanding the impact of ubiquitous computing on our everyday lives [J.8]. I will address two of those application themes, automated capture and context-aware computing, describing our results to date and the future directions in both of those areas. The other two ubiquitous computing themes, natural interaction and everyday computing, are subjects of future research. I will conclude this research statement with a discussion of practical software architecture analysis, a research theme I developed before coming to Georgia Tech and that I have been able to continue and even merge with some ubiquitous computing research themes in autoamated capture.

### Automated Capture

Much of our life in business and academia is spent listening to and recording, more or less accu-

rately, the events that surround us, and then trying to remember the important pieces of information from those events at some later time. There is clear value, and potential danger, in using computational resources to augment the inefficiency of human record-taking. This is especially true when there are multiple streams of related information in the live experience that are virtually impossible to capture as a whole through manual effort. Computational support for capture can automate the explicit and implicit relationships between related, but separately generated, streams of information. A rich record of a group interaction can support later access to aid in recalling the meaning or significance of past events.

To better understand the impact of automated capture in our everyday lives, we need tools that seamlessly capture independent streams of information, automate the temporal and semantic relationships between those streams, and provide flexible, accessible and socially appropriate interfaces to the experience. Suites of these tools can remove the burden of doing something humans are not good at (e.g., recording) so that they can focus attention on activities they are good at (e.g., indicating relationships, summarizing, and interpreting).

Our experience in automated capture of live experiences comes mainly from its application in an educational setting through the Classroom 2000 project, a system that has been in use to capture university lectures for the past three years. One of our main early contributions to automated capture research was to develop a framework for capture applications that divided the problem into four stages, pre-production, live capture, post-production and access [C.9]. We have gained an unprecedented level of experience using and evaluating this capture framework in the classroom, and in classrooms at other universities. A history of the project has been summarized in [J.7], extending preliminary evaluation results documented in [C.9] and [C.16] and techniques for the visualization and access to multiple simultaneously recorded streams of information [C.18]. A comprehensive evaluation of the impact of capture in this educational setting, based on distinguished lectures presented over the past year (see [I.1], [I.2], [I.3]) and collected from surveys of over 600 students and 22 instructors, extensive analysis of logs of system use and other evaluation techniques is currently in preparation and will be submitted for publication by the end of August 1999.

One clear outcome of our evaluations is the lack of capabilities in the access phase to support more long-term benefits. To this end, we have extended the capture and access framework to encompass the kinds of activities that occur long after a captured session and blur the line between preparation, live recording, post-production and access for a given live experience. These extensions are detailed in recent submissions discussing dynamic, searchable interfaces to classroom experiences [JS.1], anchoring collaborative discussion spaces within captured lectures [CS.1] and providing scalable visualization and control mechanisms for accessing many hours of captured experience [TR.16].

Most of our experience with capture focussed on capturing the public or shared live experience. Several researchers have looked at capture exclusively for the individual. Nobody had investigated capture in which both public and personal experiences are involved, and this is a rich research problem for both the capture and access phases. This occurs naturally in the classroom domain, as a lecturer presents public information on a topic and students personalize that information through their own notes. Initial experiments in classes attempted to support both the private and public capture, but they failed due to inappropriate private capture technology and inadequate networking (see [C.9] and [J.7] for details). We have developed a system that works within Classroom 2000 to closely integrate student notes with the public lecture experience, and this work is detailed in [C.22] and [C.26].

Beyond the classroom, we have also applied capture to informal encounters [TR.10], architectural design meetings [TR.8] and collaborative workshops. We have also begun to explore workload

issues that would help to predict the extent to which Classroom 2000 and other related media-intensive educational applications can be scaled to support an entire campus community. These workload calculations have been calibrated using live data collected from the instrumented use of Classroom 2000 over the past year, as documented in [TR.11] and [CS.2]. An interesting question which we are able to observe is how usage patterns change as students become used to the services of Classroom 2000 over time.

## **Context-Aware Computing**

The richness of human-human interaction is aided greatly by our ability to interpret the context of a given situation and adjust our behavior accordingly. Science fiction visions such as Star Trek suggest the same richness of interaction between humans and the environment, however, today we do not have easy ways of enabling the environment to sense context and wrap that knowledge into adaptable behavior. The area of context-aware computing is aimed at providing easier ways to enable ubiquitous computing applications to sense and react to appropriate cues from the people and environment. Some of the first examples of context-aware behavior sensed the location of users and adapted handheld applications appropriately according to changes in location. We built one of the first demonstrations of a tour guide application, Cyberguide, that provide location-aware cues to a traveller for indoor and outdoor tours. Several conference publications discuss this prototype (see [C.7] and [C.8]) and the entire project is discussed in a journal article [J.4].

Many context-aware computing researchers have examined location-awareness, so we wanted to explore other forms of context. In the Cyberdesk project, we demonstrated how personal productivity tools distributed across a network and on desktop and handheld devices could be automatically integrated based on the context of what single piece of information the user was focused on. For example, a name in an e-mail message could automatically launch the search for address information on that individual using any of a number of Internet-based directory services and could suggest to the user a one-step action to include that address in a personal contact database contained on a handheld personal digital assistant. One of the novel contributions in this work was an extensible architectural framework for creating inferences from a single piece of information to the variety of related information tidbits that a human would normally associate with it. The initial prototype was presented at several conferences (see [C.11], [C.12] and [C.13]) with a full discussion provided in a journal article [J.5].

Having built significant context-aware computing applications, we understood the potential power of these systems for everyday ubiquitous computing. We also understood why so few context-aware applications were making their way out of research labs and into continual use. Context-aware applications are very difficult to build and the majority of them are done as one-off prototype demonstrations that would not stand the test of time. A major reason for this is that there were very little software engineering benefits applied to the design and implementation of context-aware applications. In the past two years, we have devoted effort to applying software engineering principles to context-aware computing. The main principle is one of separation of concerns. A context-aware application gains its knowledge of the world from sensing technology. Sensors are becoming increasingly powerful and affordable, to the point where we can consider instrumenting many of the spaces we live and work in to sense critical activities. However, applications that would benefit from this sensed information from the physical world currently have to know too many of the details of physical sensing in order to use the information. We have developed a framework for separating the physical aspects of sensing from the application semantics that reacts to changes in sensed context. This Context Toolkit, first described in [C.21] creates a persistent layer, a context layer, between physical sensors and context-aware applications and greatly eases the design and evolution of the applications themselves [TR.13]. Design is eased

because the toolkit provides effective abstractions for sensed context, interpretations between context representations and aggregation of context to represent significant entities (such as people or places) from which the application programmer can develop. Evolution of an application is eased because changes to the physical sensing capabilities are shielded from the applications themselves.

As a result of the Context Toolkit, we have been able to easily modify existing applications to make them context aware. For example, we took a capture application similar to Classroom 2000 and quickly adapted it to be more suitable for informal meetings. In the classroom, the context of when a captured session occurred was fixed by a static and well-known class schedule. For informal meetings, it is never known in advance when a meeting will occur. Rather, the context of a meeting must be sensed from activity in a room or around a whiteboard, as demonstrated in the DUMMBO system [TR.10]. This work also demonstrates the close tie between context-aware computing and automated capture. In short, a capture application is one class of context-aware application in which information is stored and tagged with contextual cues (e.g., time, place, people) to facilitate later access. With this link between context-awareness and capture, we built a complex mobile capture application, the Conference Assistant, that provides capture and access features we were never able to do within the framework of Classroom 2000. This work is detailed in a conference paper [C.25] and is the subject of a journal article currently in preparation.

The major demonstration vehicle for the Context Toolkit will be to support ubiquitous sensing in a home environment, through a new and ambitious project called the Aware Home [C.24]. Thousands of sensors will be deployed to provide the information needed by various applications to know where the occupants are located within the house, what they are doing and what the appropriate mechanism is at any place and time to deliver information to those occupants. Specific applications will support care of the elderly within their own home (the Aging in Place project lead by Mynatt and Essa in the FCE Group) and the location of lost objects (Finding Lost Objects, lead by Abowd and Newstetter). In the same way that the focussed application to education in Classroom 2000 provided us great insights into automated capture, we expect the application of sensing to home life to provide a depth of understanding for context-aware computing.

## **Practical Software Architecture Analysis**

Prior to coming to Georgia Tech, I was a post-doctoral research associate at Carnegie Mellon University, where I was engaged in software architecture research. As my own and others' research and development activities confirm, the high-level design, or architecture, of a software system greatly impacts the understandability and long-term evolution of that system. In short, structure matters. One significant contribution to software architecture development was a formalization of architectural style, best summarized in a journal article with Allen and Garlan [J.1].

In collaboration with colleagues at the Software Engineering Institute, I developed a more practical technique for doing systematic evaluations of software architectures to assess their merits against a changing set of functional and nonfunctional requirements. This scenario-based evaluation technique is called SAAM (for Software Architecture Analysis Method) and has been described in conference [C.2], journal [J.2] and book [B.2] form.

The SAAM technique is a cornerstone to a large DARPA-sponsored research effort among the Software Engineering faculty in the College of Computing. My own contributions to this effort are to better support the natural architectural rationale capture activities of live SAAM sessions through use of automated capture and access technology, as described in [C.20] and to provide better input about the software architecture of a real system through a process we call architectural synthesis and described in two recent conference publications ([C.27] and [C.28]). By systemati-

cally extracting perspectives of a software system with the help of existing reverse engineering tools and then comparing the results of those perspectives, we can create through a semi-automated process a more accurate architectural representation that is faithful to the underlying implementation and its designers intentions.

## Selected Publications by Abowd

The following are a subset of Dr. Abowd's publications that are significant with respect to the research statement outlined above. A complete listing of publications can be found in Abowd's *curriculum vitae*.

### A. Journal Papers (refereed)

- [J.1] Abowd, Gregory D., Robert Allen, and David Garlan. (Significant contribution by all authors with Abowd as principal author) Formalizing style to understand descriptions of software architecture. *ACM Transactions on Software Engineering and Methodology*. 4(4):319-364, October 1995.
- [J.2] Kazman, Rick., Abowd, Gregory D., Bass, Len. and Clements, Paul. (Significant contributions and authoring by all) Scenario-based analysis of software architecture, *IEEE Software*. 13(6):47-56, November 1996.
- [J.3] Abowd, Gregory D. Software Engineering and Programming Language Considerations for Ubiquitous Computing. *Computing Surveys*, 28A(4). in *Computing Surveys*, Vol. 28A, No. 4, URL <http://www.acm.org/surveys/1996/Abowd-SEPL/>.
- [J.4] Abowd, Gregory D., Christopher G. Atkeson, Anind Dey, Jason Hong, Sue Long, Rob Kooper and Mike Pinkerton. (significant contributions from all authors, principal authorship by Abowd) Cyberguide: A mobile context-aware tour guide. *ACM Wireless Networks*. Volume 3, pages 421-433, November 1997.
- [J.5] Dey, Anind, Gregory D. Abowd, and Andrew Wood. Cyberdesk: A Framework for Dynamic Integration of Desktop and Network-based applications. *Knowledge Based Systems Journal*, Volume 11, pages 3-13. 1998.
- [J.6] Abowd, Gregory D., Anind K. Dey, Robert Orr and Jason Brotherton (Significant contributions by all; authorship by Abowd and Dey). *British VR Journal*, special issue on wearable computing, Volume 3, pages 200-11, 1998.
- [J.7] Gregory D. Abowd. Classroom 2000: An Experiment with the Instrumentation of a Living Educational Environment. *IBM Systems Journal*. Special issue on HCI / Pervasive computing. To appear October 1999. See <http://www.research.ibm.com/journal>.
- [J.8] Gregory D. Abowd and Elizabeth D. Mynatt. (equal contributions by both authors) Future directions for ubiquitous computing. *ACM Transactions on Computer-Human Interaction*, special issue on HCI research in the 21st century. Provisional acceptance to appear in 2000.

### B. Published Books and Parts of Books

- [B.1] Dix, Alan J., Janet E. Finlay, Gregory D. Abowd, and Russell Beale. (Principal contributions from Dix, Finlay and Abowd, with assistance from Beale) *Human-Computer Interaction*. Prentice Hall International, 1993. 2nd edition published in November 1997.
- [B.2] Bass, Len. Paul Clements, Rick Kazman and Gregory D. Abowd. (significant contributions by all authors, principal authorship by Bass, Clements and Kazman). Analyzing Development Qualities at the Architectural Level: The Software Architecture Analysis Method (SAAM). *Software Architecture in Practice*. Chapter 8. by Addison-Wesley. March 1997.
- [B.3] Bass, Len, Paul Clements, Rick Kazman, Gregory D. Abowd, Linda Northrop and Amy Zaremski. (significant contributions by all authors, principal authorship by Bass, Clements and Kazman). Architectural Reviews. *Software Architecture in Practice*. Chapter 9. Addison-Wesley. March 1997.

### C. Conference Presentations

#### C.1. Invited Keynote addresses

- [K.1] Abowd, Gregory D. The Impact of New Technology in Education: Classroom 2000 and Beyond. Keynote presentation at Mobility Foundation Annual Conference, Los Angeles, CA. March 1, 1998.
- [K.2] Abowd, Gregory D. Building Software for Mobile and Ubiquitous Computing Applications. Invited presentation at *IEEE CS Annual Workshop on VLSI: System Level Design*. Orlando, FL, April 16, 1998.

#### C.2. Conference Presentations with Proceedings (refereed)

- [C.1] Bass, Len J., Gregory D. Abowd, and Rick Kazman. (Significant contribution by all authors with Bass as principal author) Issues in the evaluation of user interface tools. In Taylor, Richard N. and Coutaz, Joelle, editors, *Workshop on Software Engineering and Human-Computer Interaction: Joint Research Issues*, pp. 1-12, May, 1994. Springer-Verlag will published a revised version of these proceedings in 1995.

- [C.2] Kazman, Rick, Len Bass, Gregory D. Abowd and S. M. Webb. (Significant contributions by Kazman, Bass and Abowd with Kazman as principal author) SAAM: A method for analyzing the properties of software architectures. In *The proceedings of the International Conference on Software Engineering — ICSE'16*, pp. 81–90, IEEE Computer Society Press, May 1994.
- [C.3] Kazman, Rick, Len Bass, Gregory D. Abowd, and Paul Clements. (Significant contribution by Kazman, Bass and Abowd, Kazman as principal author) An architectural analysis case study: Internet information systems. In *Proceedings of the ICSE'17 Workshop on Software Architecture*, Seattle, WA, May 1995.
- [C.4] Abowd, Gregory D., Hung-Ming Wang and Andrew F. Monk. (Significant contributions by Abowd and Wang with Abowd as principal author) A formal technique for automated dialogue development. In the proceedings of *DIS'95 — Symposium on Designing Interactive Systems: Processes, Practices, Methods and Techniques*, G. Olson and S. Schuon, editors, pp. 219–226, ACM, Ann Arbor, MI, August 1995.
- [C.5] Clements, Paul, Len Bass, Rick Kazman and Gregory D. Abowd. (Significant contributions by Clements and Bass with Clements as principal author) Predicting software quality by architecture-level evaluation. In the proceedings of *Fifth International Conference on Software Quality*. Austin, TX, October 1995.
- [C.6] Abowd, Gregory D., Rick Kazman and Jim Pitkow. (Significant contributions from Abowd and Kazman with Abowd as principal author) Analyzing Differences Between Internet Information System Software Architectures, Proceedings of ICC '96, (Dallas, TX), June 1996.
- [C.7] Long, S., Aust, D., Abowd G. D. and Atkeson, C. (Significant contributions by all with principal authorship by Abowd) Cyberguide: Prototyping Context-Aware Mobile Applications. In *Companion Proceedings of CHI'96*, Short paper, pages 293-294, April 1996.
- [C.8] Long, Sue, Rob Kooper, Gregory D. Abowd and Christopher G. Atkeson. (Significant contributions by Long & Abowd, principal authorship by Abowd) Rapid Prototyping of Mobile Context-Aware Applications: The Cyberguide Case Study. In the *Proceedings of the 2nd ACM International Conference on Mobile Computing and Networking — MobiCom'96*, November 1996.
- [C.9] Abowd, Gregory D., Christopher G. Atkeson, Ami Feinstein, Cindy Hmelo, Rob Kooper, Sue Long, Nitin Sawhney and Mikiya Tani. (Significant contributions from Abowd, Atkeson and Sawhney, principal authorship by Abowd) Teaching and Learning as Multimedia Authoring: The Classroom 2000 Project. In the *Proceedings of the Fourth ACM International Multimedia Conference (Multimedia'96)*, November 1996, pages 187–98.
- [C.10] Abowd, Gregory D. and Bill Schilit (Significant contributions by both authors, principal authorship by Abowd) Ubiquitous Computing: The impact on future interaction paradigms and HCI research. *Companion Proceedings of CHI'97*. Workshop description and overview, pp. 221–2, March 1997.
- [C.11] Wood, A., Dey, A., Abowd, G.D. (Significant contributions from Wood and Dey, principal authorship by Wood and Abowd) Cyberdesk: Automated integration of desktop and network services. *Companion Proceedings of CHI'97*, Technical note. pages 552–3, March 1997.
- [C.12] Dey, Anind and Gregory D. Abowd (principal research and authorship by Dey). CyberDesk: The use of perception in context-aware computing. In *Proceedings of the Perceptual User Interfaces Workshop*, PUI'97. Banff, Canada, October 1997.
- [C.13] Dey, Anind, Gregory D. Abowd, and Andrew Wood, (significant contributions from Dey and Wood, principal authorship by Dey and Abowd) CyberDesk: A Framework for Providing Self-Integrating Context-Aware Services. *Proceedings of the International Conference on Intelligent User Interfaces — IUI'98*. pages 47-54, January 1998.
- [C.14] Brotherton, Jason and Gregory Abowd (significant contributions by Brotherton and Abowd, principal authorship by Brotherton) Rooms Take Note: Room Takes Notes! *Proceedings of the AAAI Spring Symposium on Intelligent Environments*. AAAI Technical Report SS-98-02, pages 23-30, March 1998.
- [C.15] Mankoff, Jennifer, Jonathan Somers and Gregory D. Abowd (significant contributions by Mankoff and Abowd, principal authorship by Mankoff) Bringing People and Places Together. *Proceedings of the AAAI Spring Symposium on Intelligent Environments*. AAAI Technical Report SS-98-02, pages 168-172, March 1998.
- [C.16] Abowd, Gregory, Christopher Atkeson, Jason Brotherton, Tommy Enqvist, Paul Gulley and Johan LeMon. Evaluating the Impact of Capture, Integration and Access on Education. *Proceedings of CHI'98*, p. 440-447, April 1998.
- [C.17] Mankoff, Jennifer, Jonathan Somers and Gregory D. Abowd (significant contribution from Mankoff, principal authorship by Mankoff and Abowd) Bringing People and Places Together with Dual Augmentation. In the *Proceedings of the 1998 Conference on Cooperative Virtual Environments - CVE98*, pages 81-86, June 1998.
- [C.18] Brotherton, Jason A., Janak R. Bhalodia, and Gregory D. Abowd. (significant contribution by all, principal authorship by Brotherton and Abowd) Automated Capture, Integration, and Visualization of Multiple Media Streams. In the *Proceedings of the IEEE Multimedia and Computing Systems '98 Conference*, pages 54-63, July 1998.
- [C.19] Salber, Daniel and Gregory D. Abowd. (principal contributions by Salber and Abowd; main authorship by Salber) The Design and Use of a Generic Context Server. *Proceedings of the Second Workshop on Perceptual User Interfaces — PUI'98*, San Francisco, CA, November 4-6, 1998. pp. 63-66.
- [C.20] Richter, H., Pascal Schuchhard and Gregory Abowd (principal contributions by Abowd and Schuchhard, principal authorship by Abowd and Richter). Automated capture and retrieval of architectural rationale. Position paper to be presented at the First Working IFIP Conference on Software Architecture, February, 1999. Position paper published at <http://www.bell-labs.com/user/dep/prof/wicsa1/>.
- [C.21] Salber, Daniel, Anind Dey and Gregory Abowd. (lead authorship and principal research done by Salber and Dey. Abowd assisted in development of research ideas and writing) The Context Toolkit: Aiding the Development of Context-Enabled Applications. *Proceedings of CHI'99*, pages 434-441. Pittsburgh, PA, May 15-20, 1999.

- [C.22] Truong, Khai and Gregory D. Abowd. (principal research by Truong; joint authorship) StuPad: Personalizing the lecture experience. *Proceedings of CHI'99*, Late-breaking short paper, pages 208-209, Pittsburgh, PA, May 1999.
- [C.23] Abowd, Gregory D. Software engineering issues for ubiquitous computing. In *Proceedings of the International Conference on Software Engineering - ICSE'99*, pages 75-84, Los Angeles, CA, May 16-22, 1999.
- [C.24] Kidd, Cory K., Robert Orr, Gregory D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair MacIntyre, Elizabeth Myntatt, Thad E. Starner and Wendy Newstetter. (principal contributions and authorship by Kidd and Abowd) The Aware Home: A Living Laboratory for Ubiquitous Computing Research. *Proceedings of the Second International Workshop on Cooperative Buildings — CoBuild'99. Position paper*. October 1999. To appear.
- [C.25] Dey, Anind K., Masayasu Futakawa, Daniel Salber and Gregory D. Abowd. (significant contributions by Dey, Salber and Futakawa, principal authorship by Dey, Salber and Abowd) The Conference Assistant: Combining context-awareness with wearable computing. In the *Proceedings of the International Symposium on Wearable Computers — ISWC'99*, October 1999. To appear.
- [C.26] Truong, Khai N., Gregory D. Abowd and Jason A. Brotherton. (principal research by Truong and Abowd; principal authorship by Truong and Abowd) Personalizing the capture of public experiences. *Proceedings of the Symposium on User Interface Software Technology — UIST'99*, November 1999. To appear.
- [C.27] Waters, Bob, Spencer Rugaber and Gregory D. Abowd (principal research and authorship by Waters) Architectural element matching using concept analysis. *Proceedings of the Conference on Automated Software Engineering — ASE'99*, Short Paper, November 1999. To appear.
- [C.28] Waters, Bob and Gregory D. Abowd. (principal research and authorship by Waters) Architectural synthesis: Integrating multiple architectural perspectives. *Proceedings of the Working Conference on Reverse Engineering — WCRE'99*. November 1999. To appear.

## D. Other

### D.1. Submitted Journal Papers

- [JS.1] Pimentel, Maria, Gregory D. Abowd, Yoshihide Ishiguro, Bolot Kerimbaev and Mark Guzdial. (principal research by Pimentel, Abowd and Ishiguro, authorship by Pimentel and Abowd) Supporting long-term educational activities through dynamic Web interfaces. Submitted to *Interacting with Computers*, special issue on interfaces for a dynamic Web. June 1999.
- [JS.2] Abowd, Gregory D., Maria Pimentel and Yoshihide Ishiguro (principal research and authorship shared) Supporting life-long learning through automated capture of educational experiences. Abstract submitted to *Journal of Continuing Engineering Education and Life-Long Learning*, a UNESCO journal, special issue on Internet based learning and the future of education, June 1999.

### D.2. Submitted Conference Papers

- [CS.1] Gregory D. Abowd, Maria Pimentel, Yoshihide Ishiguro, Bolot Kerimbaev and Mark Guzdial (principal research and authorship shared equally) Integrating captured experiences with collaborative discussions. Submitted to the *Computer Supported Collaborative Learning* conference, May 1999.
- [CS.2] Chervenak, Ann L., Vivekand Vellanki, Nissim Harel and Gregory D. Abowd (principal research and authorship by Chervenak, Vellanki and Harel) Workload of a Media-Enhanced Classroom Server. Submitted to *IEEE Workshop on Workload Characterization*, July 1999.

### D.3. Software

- [S/W.1] Zen-Star 1.0. Software system for automated capture, integration and access of university lectures. Produced as part of the Classroom 2000 project. Designers: Gregory Abowd, Jason Brotherton, Christopher Atkeson and Janak Bhalodia.
- [S/W.2] The Context Toolkit 1.0. Software framework for the development of context-aware computing applications. Designers: Gregory Abowd, Anind Dey and Daniel Salber.
- [S/W.3] StuPad 1.0. Software system that works with Zen-Star system [S/W.1], to provide personalized note-taking by students in Classroom 2000.

### D.4. Technical Reports

- [TR.1] Sawhney, Nitin, Gregory D. Abowd and Christopher G. Atkeson. (principal research by all, principal authorship by Abowd) Can Electronic Notebook Enhance the Classroom? GVU Center, Georgia Institute of Technology, Technical Report, GIT-GVU-96-06, February, 1996.
- [TR.2] Abowd, Gregory D., Christopher G. Atkeson, Ami Feinstein, Yoosuf Goolamabbas, Cindy Hmelo, Scott Register, Nitin Sawhney and Mikiya Tani. (principal research by all, principal authorship by Abowd) Classroom 2000: Enhancing classroom interaction and review. GVU Center, Georgia Institute of Technology, Technical Report, GIT-GVU-96-21, September, 1996.
- [TR.3] Abowd, Gregory D. Ubiquitous Computing: Research Themes and Open Issues from an Applications Perspective. GVU Center, Georgia Institute of Technology, Technical Report, GIT-GVU-96-24, September, 1996.
- [TR.4] Dey, Anind, Lara Catledge, Gregory D. Abowd and Colin Potts. Developing Voice-only Applications in the Absence of Speech Recognition Technology. (principal contributions by Dey and Catledge, principal authorship by Dey and Abowd) GVU Center, Georgia Institute of Technology, Technical Report, GIT-GVU-97-06, February 1997.

- [TR.5] Abowd, Gregory and Irfan Essa. (equal contributions by both authors) Ubiquitous and Aware Computing. GVU Horizons article in GVU newsletter, Fall 1997.
- [TR.6] Salber, Daniel, Anind K. Dey and Gregory D. Abowd (principal contributions from Salber and Abowd) Ubiquitous computing: Defining an HCI research agenda for an emerging interaction paradigm. Georgia Tech GVU Center, Technical Report GIT-GVU-98-01, February 1998.
- [TR.7] Salber, Daniel and Gregory D. Abowd. (joint research and authorship) The Design and Use of a Generic Context Server. GVU Technical Report GIT-GVU-98-32, October 1998. Extended version of PUI 98 paper [C.19].
- [TR.8] Richter, Heather, Pascal Schuchhard, and Gregory D. Abowd. (principal research contribution by Abowd and Schuchhard; lead authorship by Abowd and Richter) Automated Capture and Retrieval of Architectural Rationale. GVU Technical Report GIT-GVU-98-37, July 1998. Extended version of ICSA1 paper [C.20].
- [TR.9] Waters, Robert, Spencer Rugaber and Gregory D. Abowd. (Principal work by Waters; principal authorship by Waters) Using the Architectural Synthesis Process to Analyze the ISVis System - A Case Study. Georgia Tech College of Computing Technical Report GIT-CC-98-22, December 1998.
- [TR.10] Brotherton, Jason A., Gregory D. Abowd and Khai N. Truong. (principal research contribution by Brotherton; Abowd and Brotherton principal authors) Supporting Capture and Access Interfaces for Informal and Opportunistic Meetings. GVU Technical Report GIT-GVU-99-06, January 1999.
- [TR.11] Chervenak, Ann L., Vivekand Vellanki, Ivan Yanasak, Nissim Harel, Roy Rodenstein, Gregory D. Abowd, Jason A. Brotherton and Kishore Ramachandran. (research contributions by all but Ramachandran, principal authorship by Chervenak) A scalable workload model of media-enhanced classrooms. Georgia Tech College of Computing Technical Report GIT-CC-99-06, February 1999.
- [TR.12] Mankoff, Jennifer and Gregory D. Abowd. (principal research and authorship by both authors) Error Correction Techniques for Handwriting, Speech, and other ambiguous or error prone systems. GVU Technical Report GIT-GVU-99-18, March 1999.
- [TR.13] Dey, Anind, Daniel Salber, Gregory D. Abowd and Masayasu Futakawa. (principal research by Dey, Salber and Futakawa; principal authorship by Dey, Salber and Abowd) An architecture to support context-aware applications. GVU Technical Report GIT-GVU-99-23, July 1999.
- [TR.14] Dey, Anind, Daniel Salber and Gregory D. Abowd (principal research by all, joint authorship) Towards a better understanding of context and context-awareness. GVU Technical Report GIT-GVU-99-22, July 1999.
- [TR.15] Jennifer Mankoff, Gregory D. Abowd and Scott Hudson. (principal research by Mankoff; principal authorship by Mankoff and Abowd) Interacting with multiple alternatives generated by recognition technologies. GVU Technical Report GIT-GVU-99-26, July 1999.
- [TR.16] Heather A. Richter, Jason A. Brotherton, Gregory D. Abowd and Khai Truong (principal research and authorship by Richter and Brotherton) A multi-scale timeline slider for stream visualization and control. GVU Technical Report GIT-GVU-99-35, August 1999.

## E. Distinguished Lectures, Panels and other Invited Presentations

- [I.1] Gregory D. Abowd and Jason A. Brotherton. Classroom 2000: An experiment in automated capture and access in an educational environment. Invited lecture at Microsoft Research, January 15, 1999.
- [I.2] Gregory D. Abowd. Classroom 2000: An experiment in automated capture and access in an educational environment. Distinguished Lecture, Computer Science Department, Brown University. March 12, 1999.
- [I.3] Gregory D. Abowd. Research experiences in automated capture applied to an educational environment. Distinguished Lecture Vassar University, April 15, 1999.
- [I.4] Gregory D. Abowd, Barry Boehm, John Knight and David Parnas. (all panelists) Panel: The relevance to industry of software engineering research in academia. *Motorola Software Engineering Symposium — SES'99*, Ft. Lauderdale, FL, June 23, 1999.
- [I.5] Gregory D. Abowd and Richard N. Taylor (panelists) Panel: Software Architecture trends. *Motorola Software Engineering Symposium — SES'99*, Ft. Lauderdale, FL, June 23, 1999.
- [I.6] Gregory D. Abowd. Panel: Technology Enhanced Learning. *Internet2 Sociotechnical Summit*. Ann Arbor, MI, Sept. 13-15, 1999.
- [I.7] Gregory D. Abowd and Anind Dey (moderators); Peter J. Brown, Nigel Davies, Mark Smith and Pete Steggles (panelists). Panel: Towards a better understanding of context and context-aware computing. *International Symposium on Handheld and Ubiquitous Computing — HUC'99*. Karlsruhe, Germany, September 27-29 1999.