

**Proceedings of the
1998 DARPA/NIST
Smart Spaces Workshop**

July 30 and 31, 1998

**National Institute of Standards and Technology
Gaithersburg, Maryland**

NIST

Editors:

Vincent Stanford

Martin Herman

Kevin Mills

Preface

The first DARPA/NIST Smart Spaces Workshop took place at the end of July 1998. This conference was made a success by all of the participants, who contributed their time to the working sessions and contributed technical vision papers printed in these Proceedings. While the individual authors are too numerous to list here, we wish to thank several individuals for their leadership roles during, before, and after the Workshop. These include: Kevin Mills, DARPA Program Manager; Martin Herman, Chief of the NIST Information Access & User Interfaces Division; Karen Sollins of MIT; Vince Stanford, the NIST Workshop Coordinator; and the Session Chairs: James Flanagan of Rutgers; Randy Pausch of CMU; Murray Mazer of Curl Inc.; Pradeep Khosla of CMU their respective NIST Session Facilitators: Vince Stanford, Sandy Ressler, Mudumbai Ranganathan, and Vladimir Marbukh. It became clear at the workshop that many elements of smart space technologies are rapidly emerging individually, but that the many unsolved problems call for coordinated research. It also became clear that there are many issues involving metrics and measurement that need to be understood in order to stimulate the coordinated emergence of these technologies.

Table of Contents

1.0 Introduction	1-1
Smart Space Scenario; V. Stanford	1-1
2.0 Keynote Session	2-1
Smart Spaces to Go K. Mills, J. Scholtz	2-3
Emerging NIST Program in Smart Spaces M. Herman	2-13
Smart Spaces: Moving into the Future K. Sollins	2-27
3.0 Situation Awareness Session	3-1
J. Flanagan Chair, V. Stanford Facilitator	
4.0 Information Appliances Session	4-1
R. Pausch Chair, S. Ressler Facilitator	
5.0 Mobility Management	5-1
M. Mazer Chair, M. Ranganathan, Facilitator	
6.0 System Integration	6-1
P. Khosla Chair, V. Marbukh Facilitator	

7.0 Invited Papers	7-1
Pseudo-IP: Providing a Thin Network Layer Protocol for Semi-Intelligent Wireless Devices K. Almeroth, K. Obrackzka, D. De Lucia	7-3
Beyond Audio-Based Speech Recognition for Natural Human Computer Interaction S. Basu, E. Jan, M. Lucente, C. Neti	7-8
AutoAuditorium: a Fully Automatic, Multi-Camera System to Televisе Auditorium Presentations M. Bianchi	7-14
The Personal Node (PN) G. Finn, J Touch	7-19
Multimodal Human/Machine Communication J. Flanagan, I. Marsic, A. Medl, G. Burdea, J. Wilder	7-20
A New Generic Indexing Technology M. Freeston	7-38
Configuration Challenges for Smart Spaces J. Heidemann, R. Govindan, D. Estrin	7-41
Enabling “Smart Spaces:” Entity Description and User Interface Generation for a Heterogeneous Component-based Distributed System T. Hodes, R. Katz	7-44
Smart Paper: Techniques for Hybrid Paper Electronic Interfaces S. Hudson	7-52
Ad-Hoc Networks and Distributed Sensing for Smart Spaces R. Iltis, F. Brewer, M. Varvarigos, J. Shynk, H. Lee, D. Blumenthal	7-57
Smart Information Spaces: Managing Personal and Collaborative Histories F. Kubala, S. Colbath, J. Makhoul	7-62
Important Technology Components in Smart Spaces C. Kwan, D. Myers, R. Xu, L. Haynes	7-68
Digital Ink: A Familiar Idea with Technological Might C. Kasabach, C. Pacione, J. Stivoric, F. Gemperle, D. Sieworek	7-76
Foldable Computing: Designing a Computer that Adapts to Your Information Needs C. Kasabach, C. Pacione, J. Stivoric, F. Gemperle, D. Sieworek	7-78
PROMERA: A Computer, Projector, and Camera All in One C. Kasabach, C. Pacione, J. Stivoric, F. Gemperle, D. Sieworek	7-80

7.0 Invited Papers (continued)

The Shadow: A Personal Experience Capture System M. Landay, M. Newman, J. Hong	7-82
Dynamic Network Computing: A vision for the Next Information Processing Paradigm R. Luhrs	7-86
Device Interaction in Smart Spaces W. Mark	7-99
Trajectory-Based Adaptation M. Mazer, C. Brooks	7-103
Mobility Management “Straw” Roadmap M. Mazer, M. Ranganathan	7-104
The Potential for Military Use of Augmented Reality Technology D. Mizell	7-110
A Framework for Intelligent Collaboratories B. Parvin, G. Cong, J. Taylor, C. Tay	7-112
Tracking Multiple, Simultaneous Talkers D. Paschall	7-117
High-Performance Tele-Immersive Active Spaces D. Reed, T. DeFanti, M. Brown, M. McRobbie, R. Stevens, M. Zyda	7-120
The New EasyLiving Project at Microsoft Research S. Shafer, J. Krumm, B. Brumitt, B. Meyers, M. Czerwinski, D. Robbins	7-126
Synthesized Multimodal Information Spaces with Content-Based Navigation J. Sirosh, M. Ilgen	7-131
The Dynamic Human Form: Wearability Issues Revealed J. Stivoric, C. Kasabach, F. Gemperle, M. Bauer, R. Martin	7-135
Adaptable Protocols for Smart Spaces M. Tsai, M. Yarvis, P. Reiher, G. Popek	7-137
Activating Everyday Objects R. Want, M. Weiser, E. Mynatt	7-140
Index to Authors	A-1

1.0 Introduction

The first DARPA/NIST Smart Spaces Conference was held at NIST North on July 30 and 31, July 1998. The purpose of the conference was to identify a community and research issues that will facilitate the creation and integration of numerous technologies that will be required to support the next generation computer-enhanced work space. It is anticipated that these technologies will cover the range of perceptive interfaces, intelligent interfaces, immersive interfaces, information appliances, mobility management, and numerous integration issues. Given a visionary scenario for the use of Smart Spaces, the workshop participants were divided into four working groups, each addressing a specific aspect of the Smart Spaces problem. These included:

I. SITUATION AWARENESS

II. INFORMATION APPLIANCES

III. MOBILITY MANAGEMENT

IV. SYSTEM INTEGRATION & INFRASTRUCTURE

A high level conceptual scenario for Smart Spaces was provided so as to stimulate the discussions in the four sessions.

Smart Space Scenario

Vince Stanford

5:30 a.m., September 2005. Federal Emergency Management Agency Headquarters (FEMA), Mount Weather, Northern Virginia.

George, a group leader with the FEMA, is currently sleeping peacefully at home, dreaming of his upcoming vacation. However, hurricane Sandy unexpectedly changed course during the night and is just now impacting the Florida coast south of West Palm Beach with one hundred and forty mile per hour winds. Power and civilian communication systems are failing under the lashing winds and rain.

George's smart office PC, Grover, receives a Priority Emergency Advisory and routes it to his pager, downloads a header paragraph and actuates its emergency management tones. George sends an acknowledgment and stumbles out of bed. "Gonna be an interesting day," he says to himself.

Presently, driving to the National Situation Awareness (SA) Center, George voice-dials Grover via a smart station embedded in his car. "Grover here, George. Ready for options, what should I do?" "Arrangements. Reserve an SA Room at my office complex for the next two days; also one near the scene." "Reserving SA Rooms, one at FEMA HQ and one at West Palm Beach. Ready for new option." "Staffing. Alert members of Team Charley and have them report to my SA Room at FEMA HQ; also contact a team at Boca Raton center." "Grover here. Alerting team Charley. Also I have confirmation on SA room six for today and tomorrow here, and a mobile Situation Awareness Center manned by Team Whiskey south of West Palm Beach in Florida." "George Signing off." "I will expect you in about ten minutes, please slow down," said Grover retrieving George's position and speed from the smart station in the car. George arrives at headquarters within a minute of the time Grover projected. Three members of Team Charley have arrived before George. They have been identified by their smart badges, speaker identification, and face recognition as they begin to set up the Situation Awareness room. The wall screen divides into work and input areas along the bottom, and opens a picture window to the mobile SA center at the top. There is a high-speed fax at the front of the SA room, which effectively allows the members of both teams to pass hard copy through the virtual windows from room to room. Field members of Team Whiskey will be wired for sound and video. Their dialog will be transcribed by speech recognizers and the feeds stored on an audiovisual server. George has a set of computer agents to filter these text streams from Team Whiskey in

the field and route the video to a window on the wall screen when critical matters develop. The SA Room is also used to send advice from George's highly qualified Hazardous materials expert, Sarah, to Team Whiskey field personnel at the scene of several hazardous material spills. She uses her area of the wall screen to view the video from head mounted cameras carried by Team Whiskey members at the spill sites and talks them through containment and neutralization procedures.

The SA room contains many sensors fused to provide capabilities that none could alone. New levels of collaborative processing are supported. The wall screen is divided into video communication areas as well as working command and data areas owned by the various personnel. These areas work in concert with task and user models to route and contextualize spoken, written, and gestured input. For example, speaker and speech recognition are combined to categorize commands according to spoken content, as well as who spoke them, and then applied to appropriate task knowledge bases. Spoken commands are also routed to the appropriate command and data areas based on speaker identity, or to other areas on explicit direction. The wall screen can thus be used collaboratively for both input and output without the use of clumsy pointing devices. The boundaries of previous-generation interfaces are transcended by sensor fusion technology.

Visual head tracking allows coordination of pen-based commands, entered using ubiquitous pads, directing them to appropriate data and command areas on the wall screen, using knowledge of user gaze to position them. Densely deployed infrared and near-field RF wireless interface and interoperability issues have been resolved long since. Each team member is a specialist who has a working session connecting him to various disaster management functions, such as supply chains for food, potable water, power, medical facilities, local police, and National Guard. These sessions were automatically routed to the SA room when the member reported for work there. Formal command phraseology and speaker recognition combine to provide good rejection of non-command speech. Speaker recognition also allows multiple speaker periods to be discarded in the SA room. Microphone arrays localize and steer beams to the working positions of the team, who are tracked as they move via video and/or smart badges. This allows good signal acquisition in the heat of live crisis management situations. Team Whiskey and Team Charley function together as smoothly as if they were in the same SA center.

As the storm passes, George remarks to Sarah that it would be nice if this room were smart enough to brush his teeth which, in the rush, he had forgotten to do eighteen hours before. The SA room later provided George with a log and summary of the dialog and activities of his personnel, at the SA rooms and in the field, for analysis and reporting.

2.0 Keynote Session

M. Herman, K. Mills, J. Scholtz, K. Sollins

The keynote session was presented in three sections and included the topics of *Smart Spaces to Go* from DARPA, *The emerging NIST Program in Smart Spaces*, and *Smart Spaces, Moving into the Future*, a report on the HP/SICS Smart Spaces Workshop in Stanford, California on July 13 and 14, 1998.

This page intentionally left blank

Smart Spaces To Go

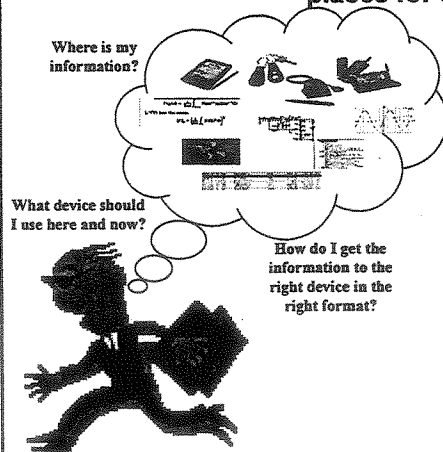
This page intentionally left blank

Smart Spaces to Go

Dr. Kevin Mills
Dr. Jean Scholtz

Smart Spaces to Go

Enable mobile workers to use multiple devices in different work places for different tasks.



New Capabilities

- Critical information follows users moving among locations
- System dynamically composes suitable multi-device, multi-mode interfaces as users move among locations
- Information adjusts interaction and presentation to devices available at each location

Integrating people with physical spaces and information spaces

Smart Spaces to Go



People work and live on the move



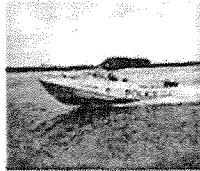
Rescue Workers



Doctors



Soldiers



Police Officers



Factory Workers



Sailors

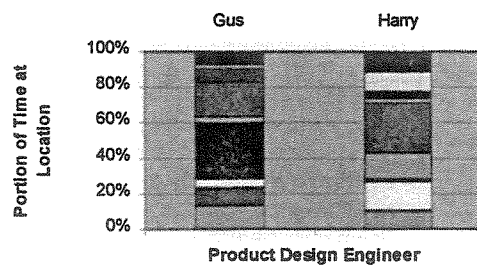
CIO for a major airline notes that 85-90% of the airline's workers are mobile information workers who spend most of their time away from their "designated" work place.

3

Smart Spaces to Go



Mobility Over Four Working Hours



In How Many Locations?

Source: Bellotti and Bly study of distributed collaboration in a product design team, Proceedings CSCW 96.

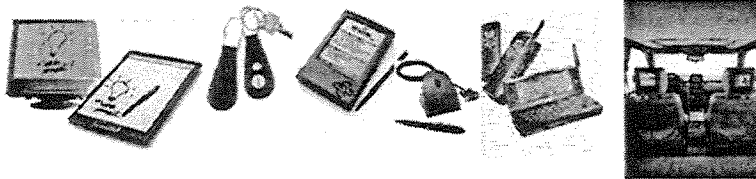
- 10-13% of work completed at desktop
- 76-82% of work spread between 11 other locations
- 8-11% of time spent moving between locations

4

Smart Spaces to Go

Today's Situation?

How do people on the go interact with information today?



Growing population of portable, embedded, wearable computing devices, each specialized for particular tasks, but

- User interacts with each device independently
- Many applications are vertically integrated with devices
- User must track, convert, and transfer information across devices

5

Smart Spaces to Go

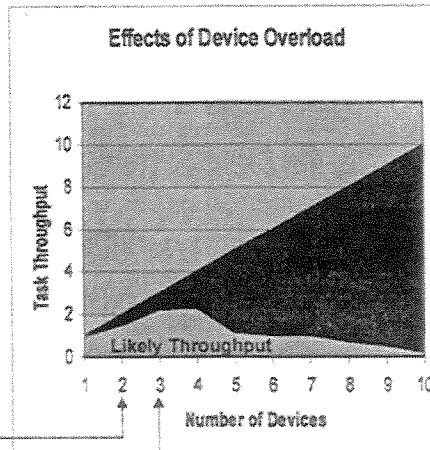
What? A Second Look

Device overload will swamp the user, even in the absence of underlying information overload.

Which devices are used most often?
Where is information stored?
How do you interact with the right device at the right time?



Average Savvy Professional
Geek and Nerd



6

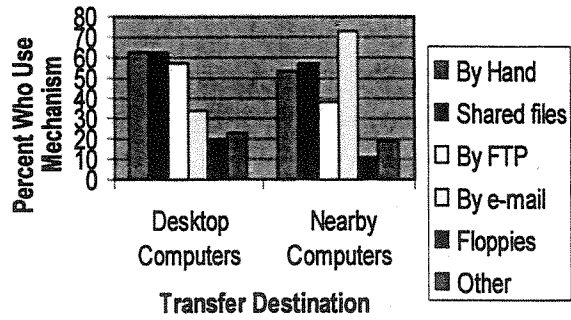
Smart Spaces to Go

More Computers, More Unproductive Time

- Computers on desktop: 54%= 3; 39%= 2; 7%= 1
- Transfer data between desktop computers: 70% very often and 25% often
- Transfer data between nearby computers: 28% very often; 23% often; 36% sometimes

Source: Jun Rekimoto, study of software engineers Proceedings of the ACM Symposium on User Interface Software Technology (UIST), 1997

How Do Software Engineers Transfer Information Among Computers?



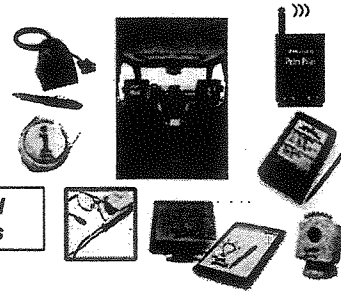
7

Smart Spaces to Go

Two Things Have Changed

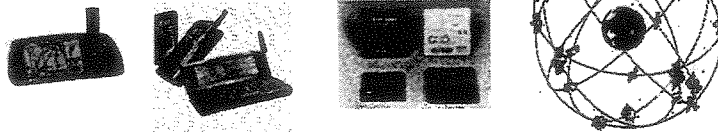
1. Networking-capable PDAs, Sensors, and Devices

IrDA and Blue Tooth Wireless LANs and Fire Wire and USB Plug-and-Play Buses



2. Location-aware Devices

GPS, Cell Phones, Active Badges



8

Smart Spaces to Go

DARPA can exploit this changing situation by developing software solutions to three hard problems:

- 1 Coordinating interactions across tens of heterogeneous devices and between seven to ten modes
- 2 Managing information mobility on a geographically significant scale
- 3 Adapting information delivery using knowledge of people, places, and devices

9

Smart Spaces to Go Coordinating Interactions

New Idea: Poly-Device, Poly-Modal Interface

A distributed coordination bus that:

- shares interaction events among networked groups of devices
- dynamically composes interfaces optimized for tasks, modalities, and devices

Decentralized Approach	Federated Approach	Centralized Approach
<i>Multicast Event Bus Announce/Listen Protocols Squawking Discovery Protocols</i>	<i>Reflector Event Bus Subscribe/Listen Protocols Push Discovery Protocols</i>	<i>Push/Listen Event Bus Registration Protocols Query Discovery Protocols</i>
MASH - UCB Visage-Link	Habanero - NCSA Orbit MAW	DISCIPLINE - Rutgers Java Beans Jini

10

Smart Spaces to Go

Tens of Devices

PDA's	Head Trackers	A/V Switches
Desktops	Projectors	Light Switches
Notebooks	Pens	Smart Cards
Large Screen Displays	Cross Pads	Active Badges
Head-worn Displays	Pointers	Speakers
Eye Trackers	Cell Phones	iButtons
Cameras	Motion Sensors	...
Microphones	Bar Code Scanners	

11

Smart Spaces to Go

Seven to Ten Interaction Modes

- Visual Display
- Audio Output
- Speech Input
- Keyboard Input
- Gaze Input
- Gesture Input
- Mouse Input
- Pen Input
- Touch Input
- Haptic Output

12

Smart Spaces to Go

Managing Information Mobility

Our Approach

New Idea: Active Information

Systems of mobile, replicable objects that communicate as groups to:

- track location, state, and trajectory of information
- track location, state, and trajectory of users
- plan information movement and replication

Multicast Tracking	Geographical Tracking	Identity-Based Tracking
<i>Context Dependent Routing Bi-directional Device Beaconsing With Intelligent Buttons</i>	<i>Location Dependent Routing Query Global Positioning With Smart Cards</i>	<i>Adaptive Routing Query Active Beacons With Smart Identity Cards</i>
BARWAN -UCB Active Services	Infostation - Rutgers Stanford Open Market	Piconet - ORL BBN

13

Smart Spaces to Go

Adapting Information Delivery

Our Approach

New Idea: Inter-Space

Couple sensor data with resource and scene description languages to model physical and logical space, so that software can:


- exploit location, proximity, visibility of resources to determine delivery devices
- adapt presentation to characteristics of available devices and services


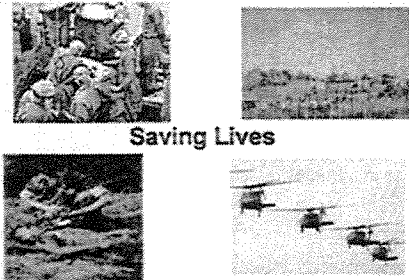
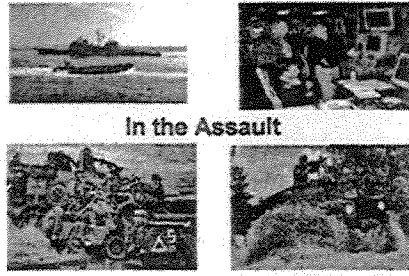
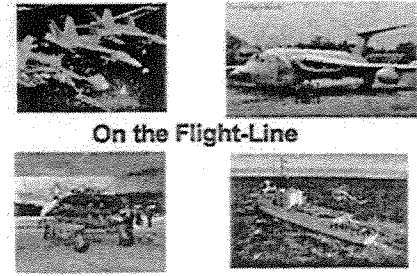
Device-Based Modeling	Image-Based Modeling	Graph-Based Modeling
<i>Embedded Device Descriptions Device Description Diffusion Proxy-based Transcoding</i>	<i>Physical Model Construction Image Sensor Mapping Visibility Algorithms</i>	<i>Adaptive Resource Mapping Graph-based Algorithms</i>
MASH -UCB Active Services	City Scanning - MIT Building Scanning - UCB	Jini - Sun Active Directory - MS

14

Smart Spaces to Go


Increased Information Availability

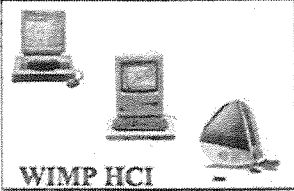
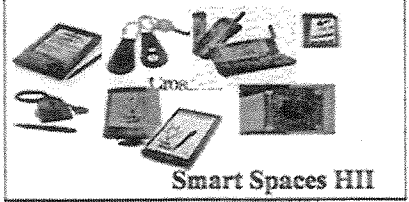


 <p>Responding to Emergencies</p>	 <p>Saving Lives</p>
 <p>In the Assault</p>	 <p>On the Flight-Line</p>

15

Smart Spaces to Go



 <p>WIMP HCI</p>	 <p>Smart Spaces HII</p>
--	---

Going Our Way?

16

**The Emerging NIST
Program In Smart Spaces**

This page intentionally left blank

EMERGING NIST PROGRAM IN SMART SPACES

**Marty Herman, Chief
INFORMATION ACCESS & USER INTERFACES DIVISION**

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

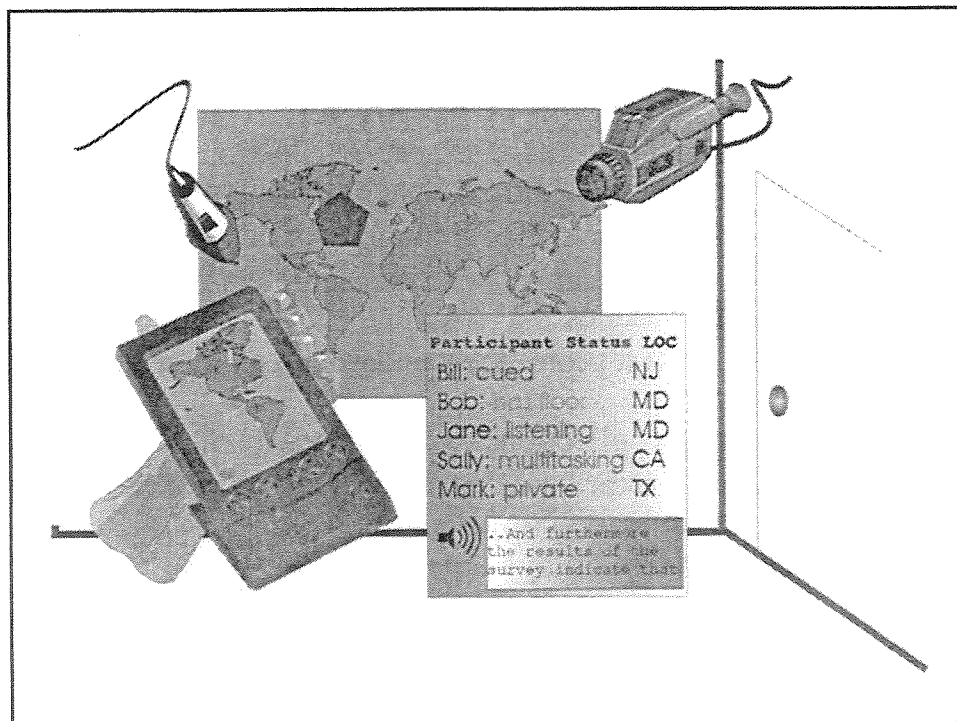
Create Smart Space Integrated Testbed at NIST

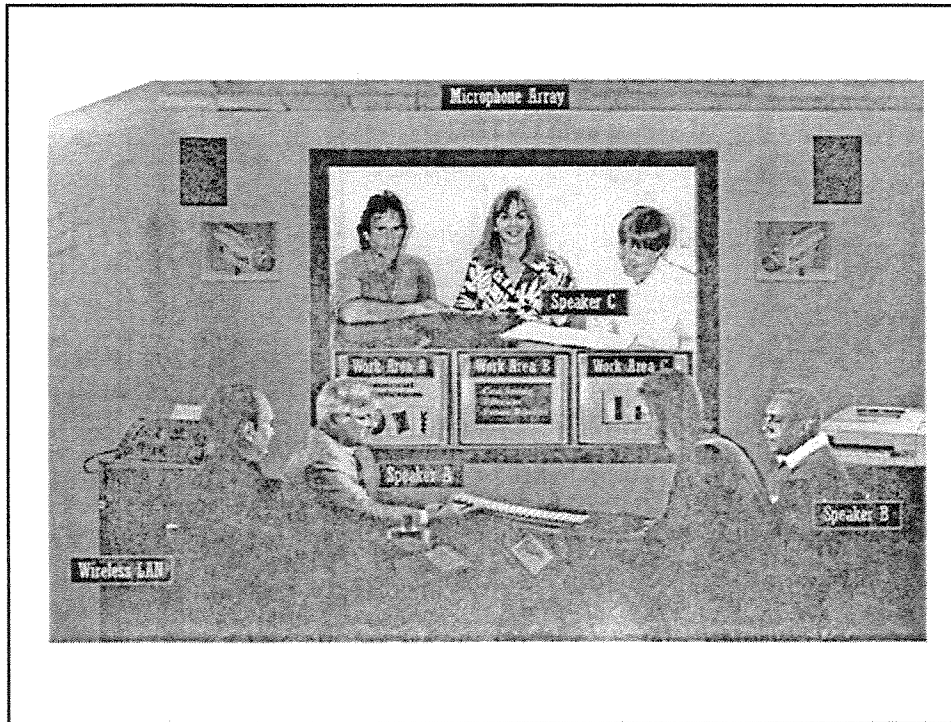
- Showcase future of Smart Space technologies
- Integrate component technologies for demos
- Develop & experiment with measurement & testing approaches
- Develop & test prototype standards (e.g., interface standards)
- Develop test collections to support evaluations
- Infrastructure for industry & academia to work hand-in-hand with NIST
- Facilitate adoption by industry

NIST Smart Spaces Testbed: Scenario

SMART MEETING ROOM

- Room understands and guides meeting participants
- Senses who is talking to whom & where they are located
- Realizes when information is requested and outputs information that seems to be useful
- Engages in dialogue with participants to get more information



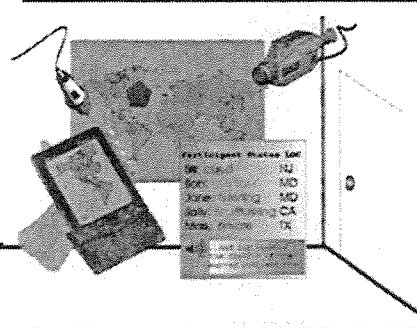


NIST Smart Spaces Testbed: Scenario (cont)

SMART MEETING ROOM

- Retrieval/extraction and presentation of multimedia information from databases and the Internet
 - Retrieval of written & spoken documents
 - Image/video information retrieval
 - Question answering
 - Information output through integrated speech and visual displays
- Recording and summarizing meeting content & events
- Collaboration within room and with geographically distributed people
 - Teleconferencing
 - Project design review
 - Distributed product design

Human Computer Interaction

	<p>Goals</p> <p>To develop measurement and test methods, and interoperability specifications, for advanced human computer interaction technologies.</p> <p>Technical Areas</p> <ul style="list-style-type: none"> • Human language technologies • Computer vision technologies • Multi-modal interaction • Information visualization • Usability engineering • Interactive tele-collaboration • Integration technologies <p>Impacts</p> <ul style="list-style-type: none"> • Increased user-friendliness of computers, leading to greater sales and acceptance by consumers • Interaction with small, embedded, mobile, & wearable computers • Pervasive use of computers at work, at play, and in people's daily lives
<p>Collaborators</p> <p>Industry: AT&T, BSN, Dragon Systems, IBM, Bellcore, Lexis-Nexis, SRI International, Claritech, Apple Computer, General Electric, Harris Corp.</p> <p>Academic: U. of Maryland, Carnegie Mellon U., U. of Pittsburgh, Rutgers U., Boston U., Massachusetts Inst. of Technology, U. of Massachusetts, Cornell U., George Mason U., New Mexico State U., U. of North Carolina</p> <p>Federal: DARPA, NSA, FBI, NIJ, NSF</p>	<p>Milestones</p> <ul style="list-style-type: none"> • FY 00: Working with industry and academia, develop measurements and tests for advanced human computer interaction technologies • FY 01: Build NIST testbeds and reference implementations for integrated human computer interaction technologies • FY 02: Work with industry and academia to apply tests to research systems to push forward the state-of-the-art • FY 03: Develop and apply approaches for testing usability of interface devices and interactive, collaborative systems • FY 04: Work with industry to develop, apply and disseminate open interoperability standards and integration technologies

Text Retrieval

Goal:

Improve information search engines

Approach:

- Annual Text REtrieval Conference (TREC)
- Large-scale test collections of documents
 - English: 5 GB of documents; 350 test questions
 - Also Spanish, French, German, Chinese
- Provides common testing mechanism
- Provides evaluation methods & measures

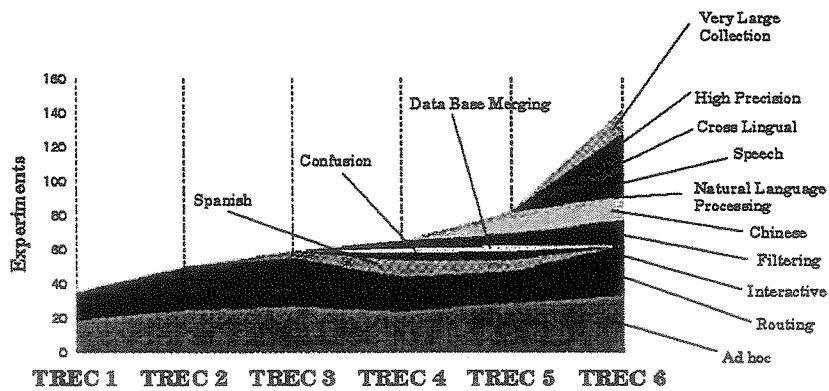
TREC-6 Participants

Apple Computer
 AT&T Labs Research
 Australian National University
 CEA (France)
 Carnegie Mellon University
 Center for Infor. Research, Russia
 City University, London
 CLARITECH Corporation
 Cornell U./SabIR Research, Inc
 CSIRO (Australia)
 Daimler Benz Research Center Ulm
 Dublin City University
 Duke U./U. of Colorado/Belcore
 FS Consulting, Inc.
 GE Corp./Rutgers U.
 George Mason U./NCR Corp.
 Harris Corp.
 IBM T.J. Watson Research (2 groups)
 ITI (Singapore)
 MSI/IRIT/U. Toulouse (France)
 ISS (Singapore)
 Johns Hopkins University/APL
 Lexis-Nexis
 MDS at RMIT, Australia

MIT/IBM Almaden Research Center
 NEC Corporation
 New Mexico State U. (2 groups)
 NSA (Speech Research Branch)
 Open Text Corporation
 Oregon Health Sciences U.
 Queens College, CUNY
 Rutgers University (2 groups)
 Siemens AG
 SRI International
 Swiss Federal Inst. of Tech.(ETH)
 TNO/U. of Twente
 U. of California, Berkeley
 U. of California, San Diego
 U. of Glasgow
 U. of Maryland, College Park
 U. of Massachusetts, Amherst
 U. of Montreal
 U. of North Carolina (2 groups)
 U. of Sheffield/U. of Cambridge
 Verity, Inc.
 U. of Waterloo
 Xerox Research Centre Europe

Text Retrieval Conference (TREC)

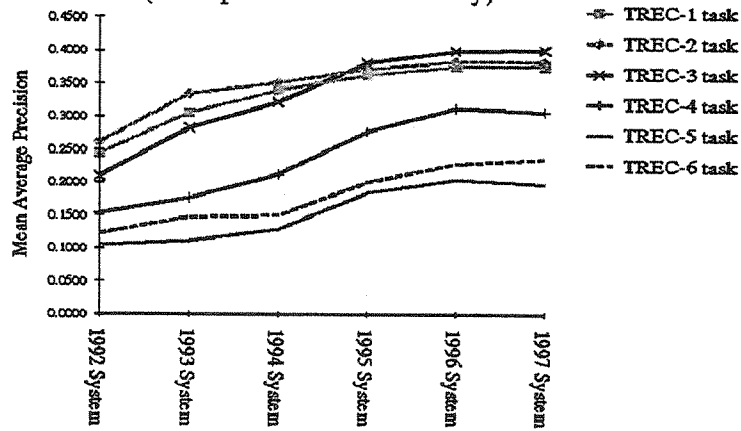
Trends in TREC



Text Retrieval Conference (TREC)

Retrieval Improvement: TREC-1 -- TREC-6

(Example: Cornell University)



Text REtrieval Conference (TREC)

Spoken Natural Language Processing

Goals:

Advance state-of-the-art in

- speech recognition and understanding
- spoken document retrieval
- speaker & language identification

Approach:

- Develop measurement methods
- Provide reference materials (speech corpora and test protocols)
- Coordinate benchmark tests
- Build prototype testbed systems

Research Community

Continuous Speech Recognition

- AT&T Bell Labs
- BBN
- Boston University
- Carnegie Mellon University (CMU)
- Dragon Systems
- IBM
- MIT-Lincoln Labs
- OGI
- Rutgers (CAIP) Center
- SRI International
- Cambridge
- Centre de Recherche Informatique de Montreal
- Karlsruhe University
- LIMSI
- Centre National de la Recherche Scientifique
- Philips

Cooperative Partners

- Linguistic Data Consortium
- CMU

Speaker Recognition

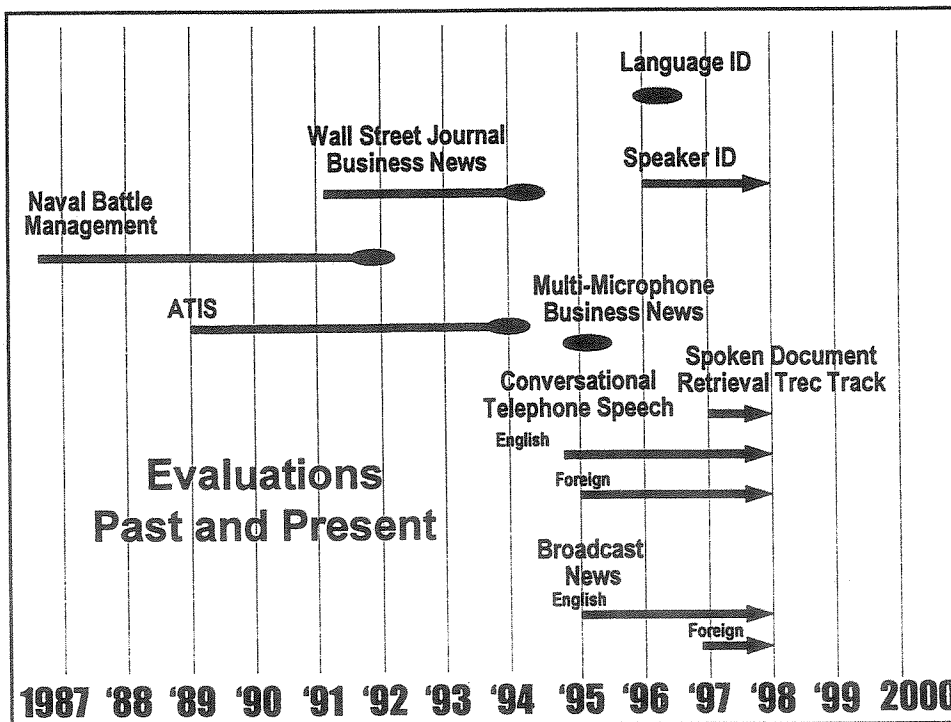
- Aegir
 - BBN
 - Dragon
 - ENST-IDIAP
 - ITT
 - MIT-Lincoln Labs
 - OGI
 - SRI
- (4-6 New Sites in 1998)*

Spoken Document Retrieval

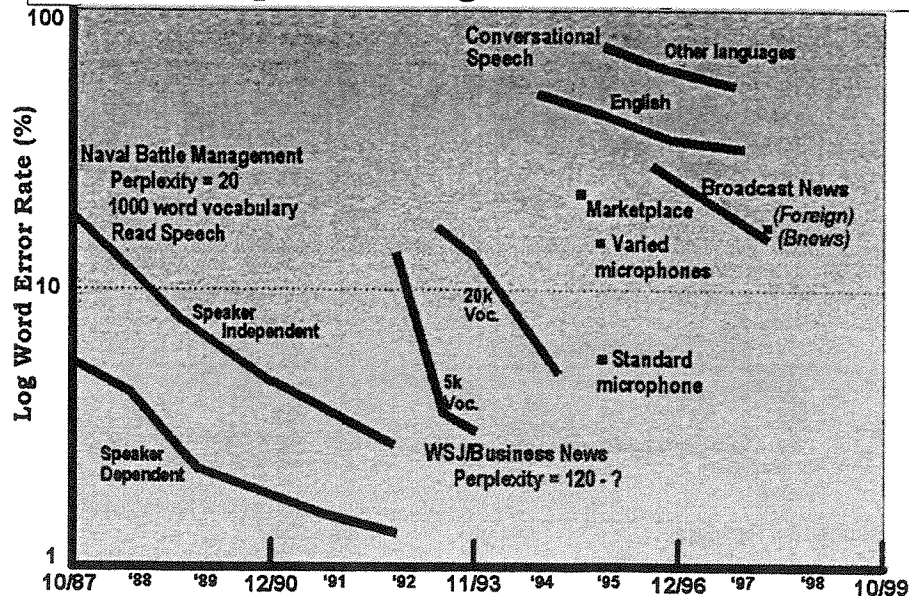
- AT&T
- CMU Informedia Group
 - Claritech
- ETH Zurich
- IBM
- Royal Melbourne Inst. of Technology
- Sheffield University
 - Glasgow
- University of Massachusetts

Retrieval Only

- City University of London
- Dublin City University
- University of Maryland
- NSA



History of Speech Recognition Benchmark Tests



NIST Smart Spaces Testbed: Technical Areas

SITUATION AWARENESS

- Speech & natural language understanding
- Mixed-initiative dialogue
- Recognizing people (e.g., using info from palmtops, speaker ID, face recognition)
- Recognizing gestures, body language
- Monitoring eye gaze
- Localizing and tracking people using sound (microphone arrays) & vision (multi-camera stereo)
- Distributed multi-sensor/multi-modal fusion (audio, visual, pen input)
- Recognizing activities, events, and intents
- Summarizing activities and events

NIST Smart Spaces Testbed: Technical Areas (cont)

MULTI-MEDIA INFORMATION HANDLING

- Information retrieval/extraction from Internet & databases
- Integrated information presentation (e.g., visualization on combination of palmtops, laptops, wall displays; speech output)

NIST Smart Spaces Testbed: Technical Areas (cont)

COLLABORATION

- Computer supported collaborative interactions
- Distributed collaboration & teleconferencing

NIST Smart Spaces Testbed: Technical Areas (cont)

NETWORKING

- Wireless communications
- Integrating portable information appliances into information infrastructure
- Projection of personal information to information infrastructure
- Wireless/wired linking of heterogeneous systems

NIST Smart Spaces Testbed: Technical Areas (cont)

INFORMATION APPLIANCES

- Display devices
- Electronic book

NIST Smart Spaces Testbed: Technical Areas (cont)

TESTING METHODOLOGIES FOR

- Speech recognition & speaker ID
- Natural language understanding
- Mixed-initiative dialogue
- Multi-media information retrieval/extraction
- Face/person/gesture recognition
- Activity/event recognition
- Event summarizing
- Visualization
- Collaboration technologies
- Display devices
- Usability of interactive, integrated systems
- Integrated system performance

NIST Smart Spaces Testbed: Technical Areas (cont)

INTEROPERABILITY

- System integration technologies
- Interoperability specifications
- Smart space integration architectures

NIST's Role

Many technology suppliers will be required to make the vision of Smart Spaces a reality.

To encourage commercial innovation, NIST should work with industry to develop open interface standards for interoperability

NIST's Role

(continued)

Smart Spaces is an emerging research area. Industry needs the following to advance research in component technologies and integration technologies:

- measurement methods
- testing and evaluation approaches
- benchmark tests
- reference materials (test data and test protocols)

**Smart Spaces Moving
Into
The Future**

This page intentionally left blank

Smart Spaces: Moving into the Future

Report on the HP/SICS Smart Spaces Workshop, Stanford, July 13-14, 1998
Karen R. Sollins

Underlying premise

Every embedded processor has a wireless transceiver and anything that can compute can communicate (Craig Partridge)

There will still be wire-based communication

Success measured in learning, finding research collaborators, and identification of architectural issues.

Three areas of discussion

- The technologies: mostly wireless networking technologies
- Interesting and challenging application domains
- Architectural issues and insights

Technologies

- Local wireless
 - Piconet (ORL)
 - Bluetooth
 - WINS (UCLA)
 - Body LAN (BBN)
- Wireless routing
 - MANET
 - Large, dense packet radio networks
- Appearing to be Ethernet
- Additional issues
 - QoS support
 - Security

Piconet from the Olivetti and Oracle Research Lab

- Goals
 - Experimental wireless network
 - Low powered
 - Low range
 - Ubiquitous
 - Systems research such as
 - High level device description
 - Ad Hoc networking
- Characteristics
 - 418 MHz architecture
 - 40 MHz bandwidth
 - Low powered (250 mW.)
 - Low range (5m.)

Bluetooth consortium

- Goals
 - Local access to data through WANs
 - Replacement of wires
 - Personal ad hoc networking
 - Universal link
 - Flexible link level security
 - Small: 1in x 1 in (down to .5in x 1 in)
 - Inexpensive (\$5)
- Characteristics
 - Range: 10 cm to 10 m.
 - Up to 8 nodes in a piconet
 - Dynamic selection of master in group
 - Up to 10 piconets locally without interference
 - Flexible link level security
 - Spectrum over range of 2.402 GHz to 2.480 GHz. In 1 MHz steps

Wireless Integrated Network Sensors (WINS) from UCLA

- Goals
 - Extremely low power: tradeoffs in communication and power
 - Deployment of sensor nets where previously unavailable
 - Extension of capabilities of embedded processors
- Characteristics
 - commercial foundry CMOS
 - 30mA drain, 3 yrs on Li coin cell
 - Measurement and RF on same 1cm device (Distributed MEMS)
 - Multihop communication

Body LAN from BBN

- Goals
 - Wireless communication for wearable devices
 - Low power
 - Low cost
 - Detection avoidance
- Characteristics
 - Short distance (body)
 - TDMA
 - Spread spectrum
 - Pre-selected master device with IP connection (perhaps also wireless)
 - Up to 100 devices in local network

Homogeneous or Heterogeneous Networking

Proposal: It all should look like Ethernet

- Step 1: make them all look the same
- Step 2: make them all look like Ethernet
- Advantage: system software will simply run as is
- Many suboptimality need study, such as:
 - Power management
 - Discovery protocols
 - Routing

Mobile Ad Hoc Network Routing (MANET) from IETF

- Assumption: Mobile constrained, but not power constrained
- Wireless, multihop networks
- Highly adaptive, efficient, scalable routing protocol
- At IP layer, above any particular wireless technology
- Supports dynamic, adaptive, context aware policies

Large Dense Packet Radio Networks

- Goals
 - Large (metropolitan area) packet radio network
 - Decentralized channel access scheme
 - Free of packet loss due to collisions
 - No per packet transmissions other than packet itself
- The model
 - Channel re-use
 - Local time-slot reservation
 - Transmission model based on Shannon bound
 - Power ratios important not absolute values

QoS support: Combining Intserv/diffserv with mobility

- How to monitor the network
 - How are things going?
 - What is around me?
- Several mechanisms needed
 - Status messages (short)
 - *Gathercast* (reverse of multicast) to collect information
 - *Transformer tunnel* including aggregator, transmitter (unicast) and dis-aggregator

Security

- Threats & vulnerabilities
 - Unauthorized access to resources
 - Breach of confidentiality
 - Access control and authentication
 - Challenge/response or tokens
 - Cryptographic access control
- Observations
 - Devices may be limited in capabilities
 - No universal standards (e.g. different legal restrictions)
 - Spectrum of requirements

Application areas

- Sensing
 - Inaccessible locations
 - Information collection
 - Sensor diffusion
- Automation
 - Factory
 - Clinical
 - Office
- Entertainment
- Location dependent applications

Application infrastructure

- Active information that monitors self
- Smart spaces as human interaction interface
- Self-adaptive modularizing apps
- "Signet ring" identification
- Introduction among devices
- Automatic synchronization
- Peer-peer communication between consenting devices

Devices

- Chair that senses direction of focus
- Cellular telephone
- "Cyberspace" that plays CD being viewed
- Light sensor
- GPS
- Combined computer, projector, etc.
- Multifunction pens & scanners
- Postcards
- Smart paper
- Foldable devices

Some specific applications

- Internet bridging
- Plain old desktop computing
- Phone that switches among office, cell, and home phones
- Monitoring engines on USS Rushmore
- Networked soldier
- Networked driver
- Smart stretcher
- Immersive communication

Architectural issues

- Self describing entities
- Need to sense the state of the environment and people within it
- Avoid surprises
- Explanation of decisions & choices
- Need a *STOP!* Button

Dimensions of an architecture

- People vs. autonomy
- Connected vs. disconnected (frequency)
- Scheduled vs. unscheduled
- Unidirectional vs. bi-directional (timescale)
- Trust: tradeoff of cost of loss - AI hard problem - cannot be resolved by computer

Architectural underpinnings

- The relationship between wired and wireless
- Reducing traffic in wireless community (power, bandwidth)
- Beaconing
- Potential asymmetry in communications
- Extremely simple device connectors with mediators as needed

Mobility in the architecture

- Mobility not wireless the issue
- Everything has unique address vs. using location for address
- Mobility doesn't imply moving to a different smart space
- Plug and play and *unplug*

More architectural issues

- Tradeoffs
 - Communication vs. sensing
 - Communication vs. processing
 - Everything vs. power
- Strategies for partitioning and remodularizing applications without human intervention
- Ad hoc smart spaces vs. architected ones

The spaces themselves

- Smart spaces may not be contiguous in cyberspace: discovering what is "nearby"
- Trust
 - Blurring of trust in physical space with trust in cyberspace
 - Transitivity of trust - erosion in transition
 - Most of the time things work correctly - finding the right boundaries
 - Tradeoff between mechanism & cost
 - Minimize surprises

Humans in the loop

- Human should be in charge (at odds with DLT position)
 - Smart spaces present choices
 - Human makes choices
- Folding together technical and policy constraints - Chooser on steroids
- *Introduction* of objects to each other

Questions

- How big are smart spaces?
- Are smart spaces smart?
- Are smart spaces spaces?
- What is the cost/benefit threshold for production of devices?
- Does instant connectivity imply wireless?
- Which functions occur at which layers?
- Is vertical integration in order to deal with low bandwidth a loss of generality?

More questions

- How do we bring real space and smart cyberspace together?
- How do we raise the level of programming above the nitty gritty of devices?
- Which of the dimensions could we tie down in order to reduce the size of the problem?

This page intentionally left blank

3.0 Situation Awareness

J. Flanagan, Chair; V. Stanford, NIST Facilitator

This page intentionally left blank

Situation Awareness In Smart Spaces



Situation Awareness Working Group
DARPA/NIST Smart Spaces Workshop
July 30-31, 1998

This presentation is the report of the Situation Awareness Working Group at the Smart Spaces Workshop.

The Situation Awareness Working group contained a variety of viewpoints and technical expertise. The resulting report is the attempt of the Group, Chair, and Facilitator to summarize a complex and fast moving discussion. The ideas are to the credit of the Group and the Chair; the inevitable omissions are the fault of the Facilitator.

We did not focus on the mobile, and in-room, networking that will be required since the Mobility Management and System Integration Sessions are expected to cover those areas. We did note that high bandwidth, local and wide area, wired and wireless networking will be a necessary prerequisite for the Situation Awareness rooms with their sensor-rich perceptive and intelligent interfaces.

Smart Space Attributes

- **Massive use of sensors** - embedded in smart spaces supporting *perceptive interfaces*
- **Recognition** - objects, gestures, speech, and people
- **Software agents** - understand activities, anticipate participants, use perceptive interfaces to form *intelligent interfaces*
- **Uses sensor fusions** - inferences; e.g.: who said what?
- **Immersive displays** - local and remote high resolution data displays
- **Metrics require:**
 - Reproducibility and transportability
 - Measurements
 - Scenarios
 - New quantitative methodologies
 - Additional scenarios - needed to capture interactions and sensor fusions
- **High bandwidth networking**

Smart Spaces, from the Situation Awareness perspective, were viewed as sensor-rich collaborative working environments. Sensor technologies will probably include microphone arrays, video cameras, smart badges, possibly IR room scanners, and position sensors. Outputs will include large screen displays and immersive acoustic imaging.

Perceptive Interfaces, using sensor fusion to recognize spoken language, particular speakers, and individual people will add new capabilities for collaborative use of existing multimedia interface systems.

Intelligent Interfaces will augment and extend perceptive interfaces drawing knowledge from sensor streams, e.g., parsing of natural language and video streams for gesture and person recognition. These will be combined with task and user models to support mixed-initiative, dialog-based, task-support systems.

Scientific Measurement will be required to evaluate the many component technologies to be integrated into the new interface paradigm. New work will be required to provide repeatable measurements for interactive systems.

Scenarios illustrating the interrelationships among the component technologies are needed to illuminate program objectives.

Network infrastructure will be crucial to the successful deployment of the Smart Space technologies.

What is a Smart Space?

- *Systems which:*
 - Identify and perceive users and actions
 - Facilitate interaction with information rich sources
 - Provide extensive presentation services
 - Understand and anticipate user needs during task performance
 - Increase:
 - rate of information exchange; local and remote
 - memory of deliberations, discourse, and decisions
 - Support collaborative use of shared data/knowledge representations

Smart Spaces employ a variety of technologies to achieve increases in the usability, collaborative accessibility, and information delivery to users. They provide improved memory of activities and deliberations for later use from the sensor . To some degree they understand and facilitate activities, and anticipate user needs as task related work progresses. They use immersive presentation services to provide more usable and understandable access to data and information based on perceived user activities and needs. They may query various information channels in anticipation of future activities.

Example Applications Enhanced by Smart Space Technology

- Analyst support, both intelligence and quantitative
- Training/education
- Tele-medicine
- Crisis management/command center
- Collaborative authorship, design and verification; local and remote
 - Documents
 - Software
 - Hardware
- Smart vehicles
- Teleconferencing as alternative to business travel

The usefulness of the Smart Space technology will be measured in terms of its effects on our ability to perform certain applications. We discussed several examples that we thought could be enhanced.

Information analysis would be enhanced by data visualization and display capabilities, by a spoken interface to data bases and query spaces, and by anticipatory queries of Web and local data bases as work patterns become clear to the agents using task models.

Training and education could make use of immersive technologies and of increased memory and retrieval of previous important events for realistic training.

Tele-medicine could be practiced using sight, sound, and touch based interfaces to allow scarce, highly-trained expertise to be “teleported” to distributed sites.

Crisis Management/Command Centers are ideal applications of the Smart Space interface technology. These will require mobile sensors and transmission capabilities discussed in the other working sessions to bring the required information to the management/command centers. The immersive display technologies allow a command/management group to coordinate the activities of field teams engaged in critical activities when combined with mobile data acquisition. Multi-sensor interfaces will offer greater ability to collaborate in the command group.

Desired Capabilities/Technologies

- Accurate, robust speech recognition for transcription in and command of the smart space
- Anticipatory Web and database query based on recognized speech
- Speech understanding:
 - Spoken dialog abstraction
 - Selective recording of minutes
- Data stream segmentation/annotation
- Face/expression recognition
- Persistence of memory, automatic links to other sessions
- Sensor fusion (acoustic/visual/other)
- Smart white boards
- Image understanding and person recognition
- Participant sensing, task identification, and adaptive response

Several new, or substantially improved, capabilities are required. We will need better acoustic signal acquisition which reduces multi-source interference, room reverberation, and background noise. Incremental improvements in speech recognition will be required. Current accuracy rates of about 85 percent have been obtained for the best broadcast news processing systems; accuracy levels not adequate for mission critical spoken interfaces. As accuracy levels increase we will want to support real time processing of recognized speech to anticipate the needs of the Smart Space users as tasks are performed. Mission critical applications will require very high accuracy levels and good robustness in terms of non-voice, and multi-voice rejection is needed. The speech and video channels can be annotated using the recognized speech and gestures, as well as speech understanding and dialog summarization. Archiving of sensor data streams gathered during critical events could allow for more complete postmortem analyses. Automatic content abstracting could be used to generate links into long-term event databases. Sensor fusion will allow correlation among data channels for annotation such as transcripts, combining what was said with who said it, or what was written or typed with who did the writing or typing. Smart whiteboards capturing drawing and writing could be a source of additional documentation of meetings. Image understanding interfaces, including visual gesture recognition and person tracking for use in fusion with other sensors, offer opportunities for high performance information processing spaces.

Emerging Technologies for Smart Spaces

- **Statistical learning algorithms:**
 - Speech recognition
 - Face recognition
 - Speaker identification
 - Blind source separation
 - Room acoustics
- **Low Cost/Power**
 - ICs - processors and memory
 - Video cameras
 - GPS
 - Network interfaces
- **Wireless megabit LANs - rapidly configurable**
- **Massive deployment of sensors:**
 - Video Camera Arrays
 - Microphone arrays
- **Immersive displays**
 - Large Screens
 - Acoustic Imaging
- **Multimedia display tools**
- **Multimedia, Object Oriented DBMS**
- **Web/search**
- **Intelligent agents**
- **Multi-view data visualization**

Statistical learning algorithms are, and will continue to be, the mainstay of Smart Space perceptive interface processing. Massive deployment of sensors will enable the pervasive, or ubiquitous, computing devices to use these sensor streams to enrich the user interfaces, and transport unprecedented amounts of information. As usual, the effect of the exponential advance of IC capability is underestimated. Sun Microsystems is showing a Java ring that has one MIP and processing capability and a two megabyte memory, which can transport digital signatures, credentials, and working set objects of the wearer. Low cost interfaces will allow "...anything that can compute to communicate". The emergence of wireless LAN technology will allow Smart Room hubs to discover and link mobile devices to the larger information infrastructure. Massive deployment of sensors is possible with individual sensors costing a few pennies: e.g., electret microphones currently available for under fifty cents today with further price declines anticipated. The profusion of data is matched by improvements in display technology, which already allows high resolution displays on screens in the ten-foot size range. Acoustic imaging for "3D sound" already available in retail operating systems and computer games. Multimedia and Object Oriented Data Bases and display tool kits are now available. Data visualization methods are under development, and will also be crucial to the summarization and processing of the volumes of information available to the smart space user.

Ideas for Smart Spaces

- Large scale smart spaces outdoors, with mobile sensors rapidly distributed
- Anticipatory web and database query based on real time dialog transcription and understanding
- Object based recording:
 - objects encapsulating audiovisuals, transcripts, etc.
 - automatic annotation of:
 - speakers
 - topics
 - attendee's history, credentials, purposes, etc.
- Collaborative interfaces based on continuous speech recognition, speaker identification, in combination with shared high resolution visual interfaces supporting group work

While Situation Awareness focuses on a central Smart Space for command, management, and collaborative design, some situations will require mobile elements acting in concert over larger spaces, such as in the FEMA scenario presented in the introduction. Wireless, dynamic, wide-area networks coupled with fielded sensors will be needed to create wide-area smart spaces.

As speech transcription, understanding and task modeling improve, it should be possible for a Smart Space to anticipate the data needs of a working group pre-query and queue information before it is actually needed to reduce latency for working groups.

Objects will be used to encapsulate audiovisual recordings along with annotated transcripts, and with display and transport methods.

There are significant opportunities to develop new generation collaborative interfaces that are not limited to the standard Windows, Icons, Menus and Pointers (WIMP) interfaces of twenty years ago.

Technologies for Use in a Demonstration Project

- Appliance discovery
- Recording
- Transcription
- Conference management
 - Logon, identification, credentials
 - Speaker identification
 - Speech recognition
 - Speech source location and camera steering

The sense of the group is that a well chosen demonstration project should be defined using a set of the technologies that were discussed. This should be implemented within a year of starting the program to uncover interoperability problems and to gain experience with actual Smart Spaces.

Key Issues for Smart Spaces

- Identify mission-critical/time-critical data
- Security/privacy/access /levels
- Positive user identification
- Sensor (recognition) confidence level measures
- Self configuring infrastructure - currently too complex for configuration and maintenance of a smart space by most users
- Enabling information management technologies:
 - Data visualization
 - Object Oriented Multimedia Data Base Management System
 - Dialog abstraction
- Integrating mobile appliances into the existing and planned infrastructure.

Key issues include how Smart Space processors might identify and prioritize information critical or relevant to the tasks undertaken in it, and how it might present these to its user group. A major set of issues in a highly networked environment with appliance discovery and automatic incorporation will be those of security, privacy and access privilege levels. Authentication of user identity may be implicit in speaker identification or visual biometrics, but for visitors to a smart space, other security protocols will be critically important. Also, the construction of even rudimentary Smart Spaces requires a high degree of skill in numerous areas. The sensor, and network software and hardware must become, as much as possible, self configuring. This exacerbates the security issues already mentioned, since it allows automatic portals of entry to the infrastructure. Fundamental issues remain with respect to how to merge the pervasive computing devices with the existing TCP/IP infrastructure. The current internet has an unacceptably high latency in propagating new IP addresses to be used for mobile, ubiquitous computing environments. One possible solution is to employ some version of Network Address Translation (NAT), or IP masquerading from Smart Space room servers. These could provide a fixed IP infrastructure and maintain communications for a dynamically changing group of devices as they enter and leave a Smart Space.

Metrics

- Mixed initiative systems still require reproducible:
 - scenarios
 - tasks
 - large corpora
- Reference data sets needed for:
 - Recognition tasks
 - Data reduction
 - Information summary
- Semantic based metrics are needed for multi-stage processes
- A Smart Space needed to record/render tasks for test materials
- Data reduction and filtering benchmarks needed
- Command/Crisis Management can offer measurable, reproducible tasks
- Test tasks must have well defined results
- Measurements for solution quality, time, and labor levels, are needed

Significant research is required in order to construct measurement protocols for mixed-initiative systems that allow multiple actions and paths. Experience in the speech recognition community has shown that well-drawn measurement programs can be very important to ongoing technical improvements in a new technology. Reference data sets will be needed for the several recognition tasks required in Smart Spaces. Some of these are already under development elsewhere in the DARPA community, but Smart Spaces can provide a test bed for integrated functionality. For example a topic detection and summarization task would use speech recognition, possibly speaker identification, and natural language parsing and discourse analysis. Measurements tasks requiring the use of multiple cascaded and parallel technologies will have to be designed. An initial Smart Space test bed will be required to record and render some of the reference materials. That is, an initial Smart Space will be needed to allow further progress to be made in the art and science of Smart Spaces; development will be iterative. It was noted by the chairman that the military mission planning community has tests and doctrine that could be incorporated into benchmarks and scoring procedures. These offer examples of how to score command team activities in terms of solution time, quality, time to solution, labor costs and other measurements.

Interoperability

- Embedding of smart spaces and mobile sensors in larger, TCP/IP based communication infrastructure
- Rapid appliance discovery and connection to infrastructure
- Dense wireless device deployment:
 - IR
 - RF
- Device protocol discovery, automatic translation among heterogeneous protocols used by multiple devices
- Security and accessibility

The group noted that there would be interoperability issues, especially in the areas of wireless IR/RF LANs. Another issue raised was that of devices acting as communication conduits for each other to establish links between incompatible devices. Protocols for network routing will also need to be investigated.

This page intentionally left blank

4.0 Information Appliances Session

R. Pausch, Chair; S. Ressler, NIST Facilitator

This page intentionally left blank

Information Appliances

NIST/DARPA Smart Spaces Workshop

Information Appliances

Definition: an object that mediates between:

the real
world and/or
people



other
computation
and storage

Attributes of Information Appliances

active v. passive

synchronous/asynchronous

touched v. activated at distance

mobile v. stationary

access latency

always with me (part of me) or not

context: devices, people, environment

intermittent v. continuous (use and/or availability)

attention: periphery v. center

stand-alone v. networked/able

Goals

Integrating people, physical spaces, and information spaces (goal for whole effort) (people can be distant, too)

Information access/use is location-aware, not location dependent.

Approaches

Use context and available modalities effectively

combine strengths of physical and digital realms

Design Tradeoffs

specificity v. generality

battery weight v. capability

part of person v. in environment

Scenarios

Medical

smart stretcher; recording of all medical care, transfusion locator for blood donor

Infinite watchman

in space & time

command watch transfer

Firefighters/Soldiers

Aggregate response

chemical warfare: vote & inject anti-toxin

drones to estimate enemy strength

instrumented firefighters

- in Central Florida brushfire scenario
- in buildings or shipboard

Telepresence

Enabling Technologies: Firefighter Scenario

rugged, wearable CPU & memory
multi-modal I/O devices & voice input
biological & environmental sensors
registering imagery w/CAD model (Hard)
position reports/combining technologies (Hard)
visual, lightweight HUD (Hard)
multi-lingual communication

Enabling Technologies: Firefighter Scenario (continued)

reliable communication (indoors)
information access
distance/aggregate sensing
automated governors (“don’t go left!”)
real-time environment modeling (Hard)
instrumented equipment/equip. modeling
3d audio peripheral displays (Hard)

Interoperability

units of measurement (generic)

precision & accuracy, including time-stamping (g)

re-purposing of information (e.g. display) (g)

existing model -> dynamic firefighter info

Specific Challenges in Firefighter Scenario

finding power shutoffs/breaker boxes

remote viewpoints via breadcrumb cameras

egress path planning/movement coordination

tagging HAZMAT containers (and people!)

lightweight and rugged

5.0 Mobility Management

Murray Mazer, Chair; M. Ranganathan, NIST Facilitator

This page intentionally left blank

Mobility Management



Mobility Management Working Group
DARPA-NIST Smart Spaces Workshop
30-31 July 1998

This presentation represents the report of the Mobility Management Working Group at the Smart Spaces Workshop.

Working Group Participants

- Ken Alen
- Kevin Almoeroth
- Gregory Finn
- John Heidemann
- Todd Hodes
- Murray Mazer
(Chair)
- M. Ranganathan
(NIST Facilitator)
- David Steere
- Roy Want

The set of participants represented a wide range of interests and expertise. This manifested itself in numerous ways, such as the wide range of applications suggested and the spirited discussions of what smart spaces were and were not.

These slides represent the attempt of the Facilitator and the Group Leader to capture the essence of the group's discussion; the group vetted these slides.

Mobility Management

- “The ability to locate, stage, and present relevant information to users as they move through Smart Spaces”
- Applications
- Mobility
- Technical Issues

The charge to the group was to discuss issues arising from “the ability to locate, stage, and present relevant information to users as they move through Smart Spaces.” This represented a focus on the mobility of the user (to the probably unintentional exclusion of other elements in a Smart Space that might be mobile); some group members objected to this focus, as we’ll see in later slides.

The group did not feel it had anything to add to what had already been discussed in the topics of Changes in Technology and New Ideas. These slides focus on Applications in/of Smart Spaces, the general topic of what kinds of things might be mobile in Smart Spaces, and numerous technical issues, including a working definition of Smart Spaces.

Applications

- **Field of Sensors**
 - spread a bunch of sensors around an area
 - assess disaster area, war zone, rainforest, ...
 - sensors fixed or mobile; people fixed or mobile
- **“Tele-operation”**
 - sensors in physical world traversed by robot
 - build virtual model for human robot control
 - “optimistic execution” with fidelity-based adjustment

One application for Smart Spaces was termed the “Field of Sensors”; the idea is to spread a bunch of sensors around an area (perhaps carefully or perhaps haphazardly). The sensors would then be used to assess the area of interest, by having the sensors self-configure into a network that could transmit its sensed data back to its targets, either outside of the area, or within the area. For example, a person might be walking through a natural disaster area receiving sensed data to guide him to a desired target, or a helicopter might fly over receiving the sensed data. Alternatively, the sensed data might be sent back to a fixed processing center. The sensors could stay where initially placed or move around the area. As we’ll see later, there are interesting design tradeoffs for the sensors.

The “Tele-operation” application concerns the remote control of mobile robots. In this application, the physical world in which a robot travels is outfitted with sensors, which contribute data to the creation of a virtual model for the human who is controlling the robot remotely. The human can use the virtual model to predictively send commands to the robot, using a form of “optimistic execution”: the robot must check the fidelity of the actually experienced physical world with the virtual model and make adjustments to the commands, to account for any discrepancies.

Applications *(cont.)*

- **Electronic Docent**
 - interact with user's system to determine preferences, adjust presentation
- **Building Information and Emergency Service (auto download of local info)**
- **Family Update Service (auto update of pre-defined status events)**
- **Highway Services Access Planning**

In the Electronic Docent application, the environment (e.g., at a museum) interacts with a system the user carries. The user's system gives preferences to the environment, which then interacts with the user according to those preferences, adjusting presentation and direction accordingly. In the museum, an expert in a certain period of art would experience the collection in a more sophisticated way than would the novice patron.

The Building Information and Emergency Service provides the user's system, upon entry to the building (such as a hotel or office complex) with information about the building, for later use. The information may be mundane (location of washrooms, business-related (location of meeting), convenience-related (location of printer), or life-saving (location of emergency exits).

The Family Update Service represents a class of applications in which the user defines a set of events that she wishes to be sensed and reported. For example, the traveler may wish the system to notify his family that he has arrived at Dulles Airport (perhaps an hour late) and then that he has arrived at the destination hotel (perhaps after a delay at the car rental agency). The environment through which the user travels can sense user-defined events and report them to user-defined targets.

The Highway Services Access Planning application assists the traveller in deciding at which points to access services, based on forward sensing, such as distance, match with desired service properties, queue lengths, etc.

Types of Mobility (not just the user)

- Human user
- Mobile devices (of user, of smart spaces)
- Application context of user
- Infrastructural properties
- Smart Space itself
- Some components move themselves, some moved by an external agent

The Working Group statement of purpose focused on the mobile human, but other things may be mobile in a Smart Spaces World. These include devices, of the user (such as laptop, watch, belt buckle, vehicle, light switch with favorite settings, etc.) or of the Smart Space. The user's application context may move to accommodate the user (this may include application, data, network context, etc.). Aspects of the infrastructure may travel--this may include physical aspects or virtual aspects (e.g., a Smart Space that needs a certain functional capability may download it from another Space). The Smart Space may itself be moving relative to the surrounding space, which might be other Smart Spaces or "dumb spaces." For example, the user's vehicle may provide a Smart Space that, while travelling on a highway, is immersed in a relatively dumb space that has intermittent portals of connection to more sophisticated capabilities.

The mobile elements in a Smart Space world may move themselves, or they may be moved by an external agent.

What is a Smart Space?

- No consensus on set of defining characteristics
- Boundaries? Definition? Static vs. dynamic?
- Can Smart Spaces overlap? Is there just one?
- Can they be in a hierarchy?
- Can disjoint Smart Spaces communicate?
- “A Smart Space enables the user to purposefully interact with a physical environment integrated with an information environment, often with the ability to modify one based on the other.”

The Group’s members had diverse ideas of what the term “Smart Space” referred to. After lengthy and wide-ranging discussion, the group did not reach consensus on a set of issues, including defining characteristics for Smart Spaces, whether there is *one* Smart Space, whether there are multiple spaces which may overlap, whether Smart Spaces may be arranged hierarchically, how a Smart Space’s boundaries are defined, whether they are static or dynamic, whether disjoint Smart Spaces may communicate via traditional means (such as IP), etc.

The group was willing, however, to subscribe to the following statement about Smart Spaces:

“A Smart Space enables the user to purposefully interact with a physical environment integrated with an information environment, often with the ability to modify one based on the other.”

Interoperability

- Need ways of making systems interoperate and devices operate in different infrastructures.
- Key enabling technology: Need an interoperability (“spanning”) layer at the network and object interface level.
- What is the cost of interoperability?

Given that the elements of Smart Spaces will be created and deployed by a diverse group of technology developers, it is critical to establish means for different systems and devices to interoperate. The key enabling technology to support interoperability was seen to be a “spanning layer” that provides network and object interfaces that each element may implement in its own way but that provides a means for interaction.

Some members of the group argued that interoperability was *not* a desired property of all elements in all Smart Spaces, primarily because of the cost of supporting interoperability and the tradeoffs that would ensue (e.g., the cost of a sensor might rise because of the added protocol support).

Interoperability *(cont.)*

- Avoid vertical, non-interop systems: value in ubiquity and aggregation of multiple
- Lesson of Internet: flourish through broad interoperability and modularity
 - limit point of interoperability to specific place
- What are similar bases of interop for Smart Spaces?

Why does one want interoperability? To avoid the spectre of vertical, non-interoperable systems being deployed. Rather, there was unanimous agreement in the value of enabling the ubiquitous deployment of Smart Spaces as created by multiple innovators.

The lesson of the Internet was raised as a guiding principle: flourish through supporting broad interoperability and modular interfaces (limit the point of interoperability to a specific place, permit innovation above and below it).

Interoperability (*cont.*)

- “Spanning layer”
- Network protocol for communication
- Shared description mechanisms (for devices and service interfaces) to permit meaningful adaptive interaction

The “spanning layer” of interoperability for Smart Spaces was seen to involve both the *network layer*, for communication, and *shared description mechanisms* (for device and service interfaces) to permit meaningful adaptive interaction.

Interface and Configuration Discovery

- Need a programmatic interface to elements in physical space
- Mobility infrastructure must support location-dependent resource discovery
- How to discover:
 - downloadable based on unique ID
 - assumed
 - discovered from device
 - discovered from proxy or space broker

There was agreement on the need for programmatic interfaces to elements in physical space; that is, for the controllable elements in a physical space, it must be possible for programs to discover the relevant interfaces (and current configurations) and exercise them in order to control those elements. Also, the infrastructure must support location-dependent resource discovery (“which light switches control *this* room? what are their current states?”).

Discovery may be achieved by a number of means, including: download (e.g., via HTTP) based on a unique ID of the element of interest; assumed (not very flexible); discovered from the device itself; or discovered from a proxy or “space broker.”

Interface and Configuration Discovery (cont.)

- How to name things in physical space in order to ask for information about them?
- Must support interface (reflective) queries including queries like:
 - What are the capabilities of the device?
 - What interfaces does it support?
- Exploit existing infrastructure (such as HTTP etc.) whenever possible.
- XML, CORBA interfaces, Java RMI

One must be able to name the things in physical space in order to discover their properties.

There was agreement that elements (both physical and virtual) in the space must be able to support interface (or “reflective”) queries, such as “what are your capabilities” and “what are the interfaces for control?”

The group agreed that the chosen means should exploit existing infrastructure whenever possible (such as HTTP).

The group discussed several specific mechanisms, include XML, CORBA interfaces, Java RMI, and others.

Identifying Relevant Information

- Defining events of interest and how to sense them
- To whom to report the sensed event
- How to aggregate them
- How to report/route the events to the target
 - directly or through intermediary service?
- How reliable must sensing and transmission be?

For applications in which the infrastructure must sense specified events, there is the problem of defining the events of interest and how to sense them. Also the targets of the sensed event must be identified, as well as any desired aggregation and condensation techniques to be applied to the events. Means for reporting the events to the target must be identified (and these might be directly to the target or through intermediary services).

Finally, there is the question of how reliable sensing and transmission must be? Does the application require that all sensed data and events reach the targets reliably, or may the system use a 'best-effort' satisficing approach? The answer to this question affects a number of other issues, including the required capabilities of the network.

User-centric Adaptivity

- Infrastructure that supports user- and task-relevant data selection and presentation
 - firefighter, general, physician
- Model of user roles, tasks, prefs., priorities
- General system notions of quality of service, resource management and contention
- Choosing the data and how to present them on available interfaces more than staging

One of the goals of the Smart Space is to support adapting to the informational and physical needs of the user. This requires an infrastructure that supports user-relevant and task-relevant data/event selection and presentation sensitive to physical and informational context. For example, a firefighter may need to know the means for fighting the specific fire she's facing; the general may receive higher downlink bandwidth than the lower-ranked personnel in a shared conference space; the physician reviewing CAT scan results may receive service via the highest resolution display available in the current location.

These capabilities require models of the users, their roles, organizations, tasks, preferences, priorities, etc.

This also relates to general system notions of quality of service, and resource management and contention.

There was a feeling in the room that the issues around choosing the relevant data and presenting them on available interfaces had more challenges than did the staging of those data.

Location Information

- Of what? To what granularity?
- Mix of techniques (learn from experience)
- Who beacons, who listens (user/Space)
- What kind of queries
- Privacy
- Distribution of location information
- different levels of information accuracy (e.g., inside building vs. outside)

The issue of acquiring and using location information is filled with interesting issues, including:

- the location of what? to what granularity (i.e., to what physical granularity (a room, a chair, a building, etc.) and what granularity of entity may be located (e.g., a human, a lap-top, a light switch, an ant, ...).
- acquiring location information will likely require a variety of techniques including RF, IR, GPS, etc. Those with experience with Active Badges related the deficiencies experienced.
- Does the trackee beacon, or does the tracker beacon?
- What kinds of queries are supported? What kind of privacy constraints?
- It is seen as inevitable that the location information system and its clients have to deal with different levels of information accuracy.

Metrics

- Specific metrics for specific system components
- The key indicators are on human terms:
 - Ability to accomplish function safely.
 - Accuracy
 - Having the smart space work as intended.
 - Systematic user evaluations.
- Programming ease.
- Scalability.
- Establishment of an industry-accepted standard for smart spaces.

The group did not attempt to discuss specific metrics for specific system components, except to agree that those metrics must be defined relative to the components themselves.

However, the primacy of the human in Smart Spaces was acknowledged through the suggestion that some important metrics are measured on human terms, such as the ability to accomplish a function safely and with sufficient accuracy, and having the Smart Space work as intended. Evaluating these kinds of metrics requires systematic user evaluations.

Another human issue is the ease of programmability of the various elements.

Another issue is how well a given Smart Spaces technology scales.

Someone suggested that a major hurdle will be overcome with the establishment of an industry-accepted standard for Smart Spaces--at that point, it will become clear that the technologies involved have been judged to have meritorious (or at least deploy-able) properties.

Sensor Issues

- **Sensor-based information**
 - sensors must self-configure into a network
 - avoid swamping (perhaps impoverished) net
 - gathercast, aggregation
 - how reliable must the gathered data be?
 - Affects the network support; affects device properties (i.e., more reliable data delivery requires stronger network properties)
 - high redundancy and unreliable communication to satisfy?

There was some detailed discussion about the nature of sensors in Smart Spaces. For many applications (such as Field of Sensors and Tele-operation), the sensors must configure themselves into a network capable of carrying the sensed data to its targets. The transmission of data must avoid swamping the available net, suggesting techniques such as gathercast and aggregation. This is especially important since the sensors may themselves be “thin” devices, suggesting that the resulting net will be impoverished.

The issue of how reliably data must be sensed and sent to the targets arose again as a key tradeoff issue in the design of sensors.

Sensors (cont.)

- Have to be cheap.
- Have to be programmable.
- Have to support resource discovery and interface discovery.
- Computational power may be limited.
- Communication capability may be constrained (may only support one-way communication).

There was strong sentiment that sensors must be inexpensive to build in large quantities (this is especially true for Field of Sensors type of apps). There will be difficult tradeoffs, involving diminished computational and communication capabilities.

Resource discovery interface has to support queries like:

- what is this device?
- what can it do?
- how much does it cost to use?

Sensors may have very limited computational resources and hence may not be able to support a full fledged communication stack. The group discussed some applications that can use one-way communication in Smart Spaces.

Network

- What capabilities required?
- Diverse, depending on devices and context
 - sensors, smart airport, smart conference room, ...
- Where is interoperability achieved?
- Not clear can assume IP for some devices
- Affects ways in which information can be gathered

There was discussion on what protocols are needed for network communication in a smart-space environment. Due to the limited computational and memory capacity of sensor devices, it was agreed that it would not always be possible to assume that a full IP stack can be provided in the sensor. Thus, new, lighter weight protocols may be necessary.

More Network

- One-way protocol
 - for what class of apps?
- Addressing:
 - specific device
 - geographical addressing
 - general property-based addressing
- Communicate with all devices, or class of devices with proxy support

There was some discussion of possible applications that can be deployed in the face of one-way networking.

There was also some discussion of addressing capabilities required for Smart Spaces, including the ability to communicate with a specific device, with devices in a given geographical area, or with devices matching certain specified properties.

There was also some discussion of the need to be able to communicate directly with all elements, vs. the (likely) possibility that some classes of devices will use proxy support.

Other Issues

- Composing individual services into larger-grained services
- Importing capabilities into Smart Spaces
- Composing interactive interfaces to individual and composed services
 - adaptively creating interface to Smart Space
 - linking interface components to physical elements
 - adapting existing interface

6.0 System Integration

P. Khosla, Chair; V. Marbukh, NIST Facilitator

This page intentionally left blank

Systems Integration Group -- Smart Spaces Workshop

Son Dao

Ron Iltis -- iltis@ece.ucsb.edu

Chiman Kwan -- ckwan@I-a-I.com

Richard Luhrs -- richard.a.luhrs@cpmx.saic.com

Peter Lucas-- lucas@maya.com

Karen Sollins -- sollins@lcs.mit.edu

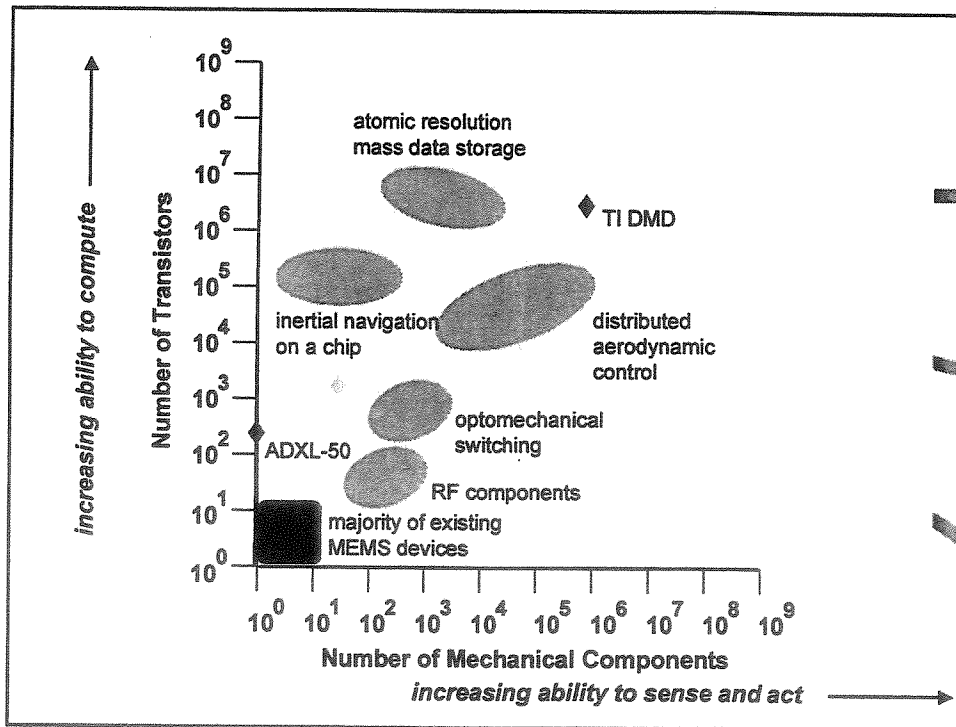
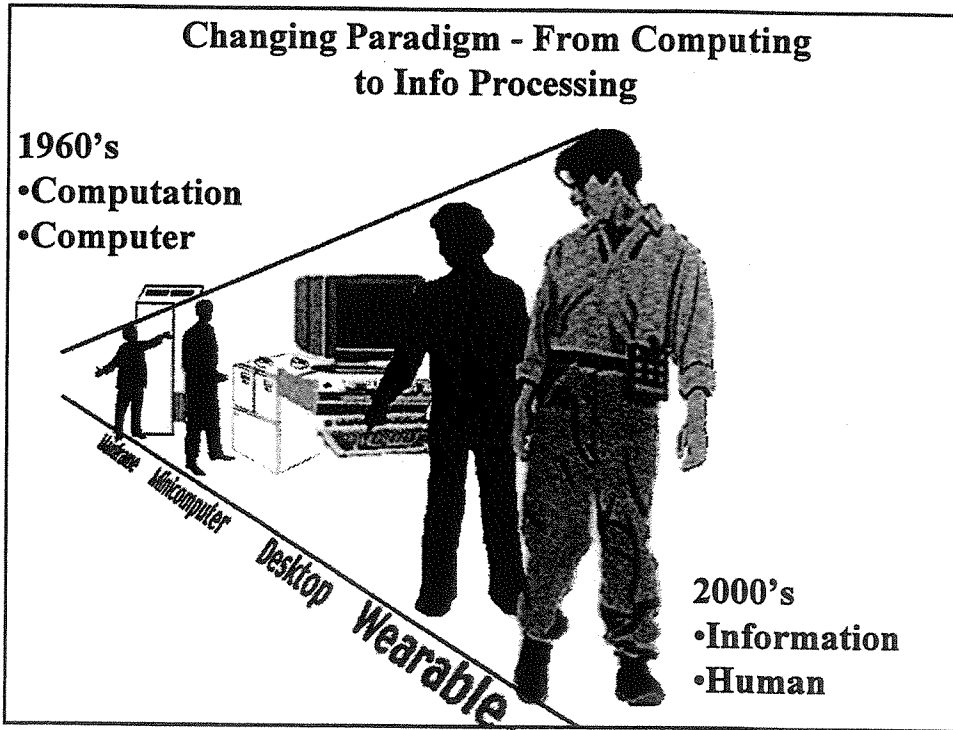
Bill Mark -- Long Pole

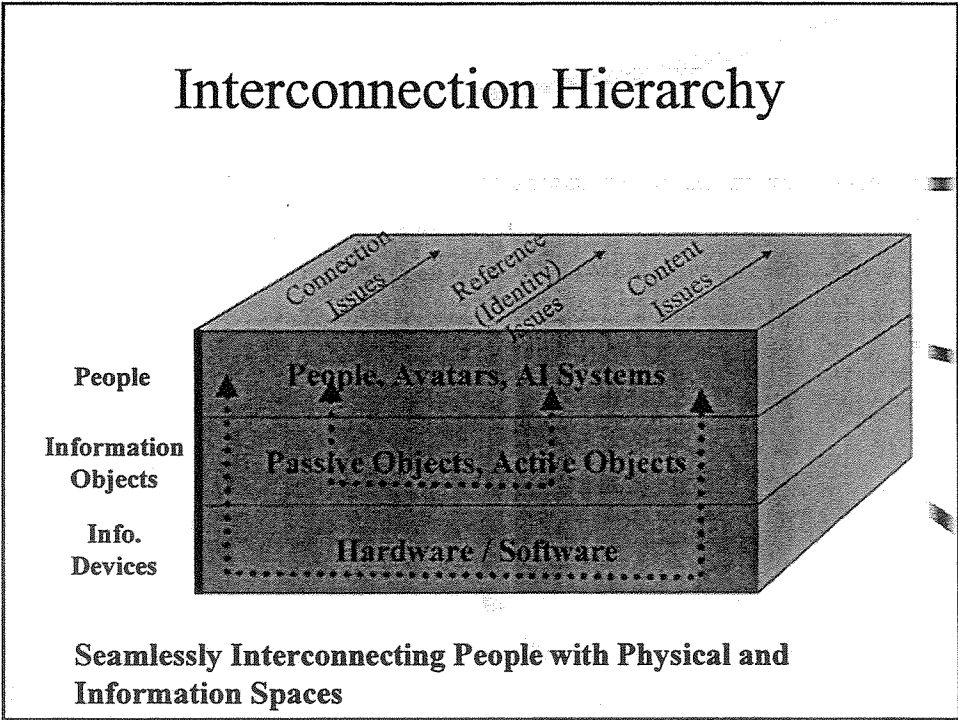
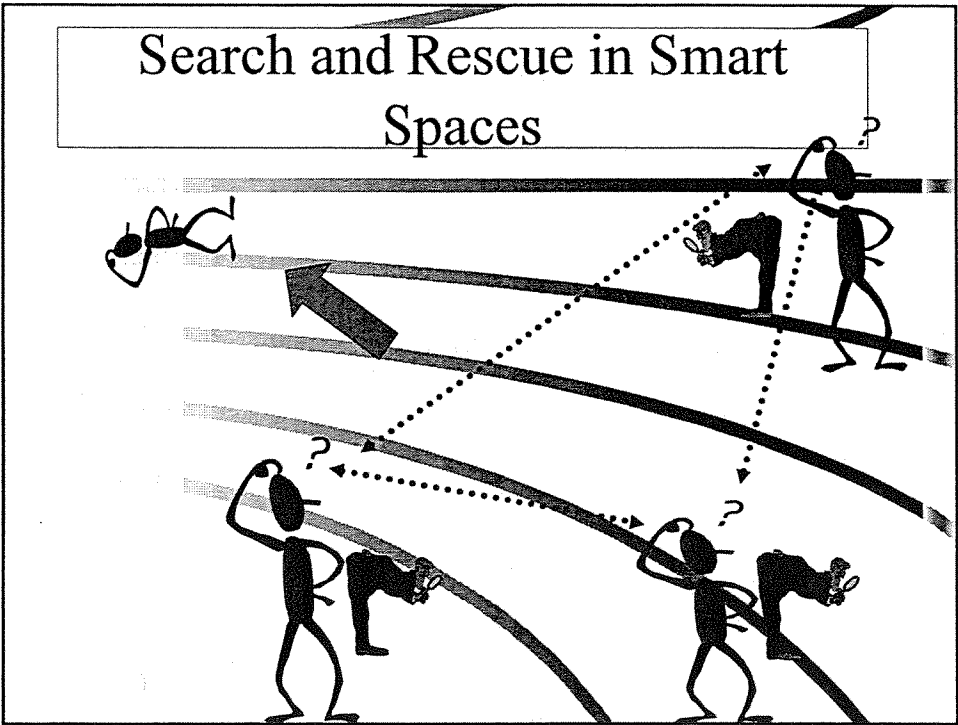
Pradeep Khosla

Vladimir Marbukh -- marbukh@nist.gov

State of Technology - Why Now?

- **2B micro-processors manufactured in 1996**
 - **95% of these are Embedded processors**
- **Communications**
 - **Wired Connectivity (band-width) nearly there**
 - **capacity for more than a few Mb/sec per person**
 - **Wireless Infrastructure (nearly) in place**
- **Info Delivery Devices (Wearable “X”) available**
- **MEMS enabling “computational sensors”**
- **Storage is cheap**





Technical Capabilities

- **Connection**
 - how do devices start interacting?
- **Reference**
 - what do devices call each other?
- **Content**
 - what do devices talk about?

Research Issues

- **Architectures for Heterogeneity**
- **Configuring Software for Smart Space**
 - “Beans of the future” will come together to create applications
- **Dynamic Aggregation of Information**
 - how does info (collected locally and independently) get aggregated to create a shared picture
- **Methods for creating “Trust”**
 - Password
 - Familiarity, ... etc.

Research Issues

- **Networking can enhance Smart Spaces**
 - Need improvement in existing MultiUser connections (in Wireless)
 - Quasi Synchronous CDMA
- **Interaction Horizon is an important issue**
 - Task and Capability Dependent
 - Optimizing throughput through where/how/how-much to compute

This page intentionally left blank

7.0 Invited Papers

A variety of papers were submitted by the Workshop participants. These address the research topics that were identified by the individual researchers or groups and form our basis for an emerging understanding of Smart Space functionality and components.

This page intentionally left blank

Pseudo-IP: Providing a Thin Network Layer Protocol for Semi-Intelligent Wireless Devices

Kevin C. Almeroth
Dept of Computer Science
University of California
Santa Barbara, CA 93106-5110
almeroth@cs.ucsb.edu

Katia Obraczka
Information Sciences Institute
Univ of Southern California
Marina del Rey, CA 90292
katia@isi.edu

Dante De Lucia
Computer Science Department
Univ of Southern California
Los Angeles, CA 90089-0781
dante@usc.edu

ABSTRACT

In the near future users will be able to move freely and still have seamless network and Internet connectivity. We envision that the Internet of the future will interconnect mobile or stationary clouds into the existing IP infrastructure. While many of the Internet protocols are immensely successful in traditional networks, we believe they will be inappropriate for communication among limited-capability devices in amorphous clouds. The question we are trying to address is whether the network layer services provided by IPv4 and IPv6 [1] are necessary and sufficient for supporting heterogeneous devices in these highly dynamic, arbitrarily dense environments. The problem with an IP infrastructure is that, for certain applications, it adds unnecessary complexity. The proposed research is based on the specification of a new network layer protocol for intra-network communication in clouds containing devices with limited processing and communication capabilities. Our *Pseudo-IP* protocol is designed to operate among devices in the farthest branches/leaves of an internet while providing inter-network connectivity with other clouds and the existing IP infrastructure. Our goal is to extend the scope of IP to environments containing devices that cannot handle the extra complexity introduced by routing, error detection/recovery, optional headers, and even addressing. More intelligent devices interacting in the cloud will be responsible for interoperating with the existing IP infrastructure including IP-based devices that happen to be roaming locally.

1. Introduction

In the near future users will be able to move freely and still have seamless network and Internet connectivity. Portable computers and hand-held devices will be for data communication what cellular phones are now for voice communication: they will keep users connected at all times. In addition to being continually connected over time, the concept of “universal connectivity” also means that a variety of “unconventional” devices will be connected to the Internet. A variety of devices, including sensors, home appliances, light switches, etc., will be interconnected forming *clouds*. We envision that the Internet of the future will interconnect these (mobile or stationary) clouds and the existing IP infrastructure. As users roam among clouds, they will encounter, and be required to communicate with, a range of devices varying in processing, mobility, and communication capabilities.

While many of the Internet protocols are immensely successful in traditional networks, we believe they will be inappropriate for communication among limited-capability devices in amorphous clouds. The question we are trying to address is whether the network layer services provided by IPv4 and IPv6 [1] are necessary and sufficient for supporting heterogeneous devices in these highly dynamic, arbitrarily dense environments. The problem with an IP infrastructure is that for certain applications it adds unnecessary complexity and overhead. One example scenario is a cloud of thousands of sensors transmitting small pieces of data. The data portion, assuming traditional IPv4 or IPv6 headers and lower layer protocol headers, will be a very small percentage of the overall packet size. Besides the inefficiencies of payload bytes versus header bytes, there is also the issue of the devices' limited processing and communication capacity. The numerous sensor devices may simply transmit their data and the overall system would rely on a roaming user carrying a more intelligent device. Alternatively, the sensor cloud could contain one or more data collection devices the roaming user's computing device is able to locate and query. Whatever the scenario, there will likely be devices with limited functionality that still have important data to communicate.

The premise of this paper is to introduce a new network layer protocol for intra-network communication in clouds containing devices with limited processing and communication capabilities. Our *Pseudo-IP* protocol is designed to operate among devices in the farthest branches/leaves of an internet while providing inter-network connectivity with other clouds and the existing IP-based Internet infrastructure. Our goal is to extend the scope of IP to environments containing devices that cannot handle the extra complexity introduced by routing, error detection/recovery, optional headers, and even addressing. More intelligent devices interacting in the cloud will be responsible for interoperating with the existing IP infrastructure including IP-based devices that happen to be roaming locally.

While there has been a great deal of interest in wireless pro-

ocols, much of the work has focused on providing higher bandwidth and more sophisticated communications services. Our approach is directed towards simplification. We believe that our Pseudo-IP protocol will be an important enabling technology for the Internet of the future since it will allow limited-capability devices to be connected to the existing IP infrastructure.

One research initiative related to the proposed ideas is the Daedalus/BARWAN project [2] at UC Berkeley. More specifically, they have proposed an architecture that supports adaptive client device's functionality to new services that are discovered/located as the client moves [3]. Their architecture is based on the existing IP infrastructure. Other research initiatives in this area are beginning and should be jump started by DARPA interest and funding.

The remainder of this paper is organized as follows. Section describes a number of potential applications and their requirements. Section gives an overview of the concept behind Pseudo-IP. Section summarizes the research challenges and how they relate to our Pseudo-IP protocol.

2. Applications and Requirements

The motivation for Pseudo-IP is grounded in a number of cloud-based applications and scenarios. We describe some of these scenarios and their specific protocol requirements below.

- **Home spaces:** Homes of the future will be full of devices ranging in intelligence from dumb to semi-intelligent to fully-intelligent devices. Some examples include dumb devices like light switches; more intelligent appliances with embedded circuitry like TVs and microwaves; and programmable, network-aware devices such as PCs. The least intelligent devices may only be capable of broadcasting state information. Some may have additional functionality which allows them to receive and process commands and then change state. The most intelligent devices would be responsible for state collection, device control, and management. Consider the commonly referenced example of a fully connected home. All electronic devices are connected together in a controllable network. Voice commands are detected, translated, and executed through some voice recognition system. Many household devices will have wireline connectivity but some may be wireless. In either case, these devices must be capable of receiving, executing, and acknowledging commands. Another important ser-

vice required in this scenario is security. It will likely be an important issue, especially from from the need to authenticate commands and prevent unauthorized users outside the home from controlling devices in the home.

- **Highway spaces:** Highway environments offer a slightly different set of requirements from the home environment. Objects in the highway environment are likely to be more intelligent but may be more transient. Current projections suggest vehicle-based communications systems providing services other than cellular telephony will be able to communicate using IP. So in several of the highway scenarios, many of the key objects will be capable of speaking IP. But because of the limited range capabilities of these devices, an intra-cell network protocol will likely not require IP-style services. One common application consists of drivers receiving information about road conditions, restaurants, shopping, service plaza amenities, etc. via broadcast transmissions. This type of data can be broadcast repeatedly so reliability (other than through non-ACK based techniques like forward error correction[4]) is not required. However, some applications may provide transaction-based services like ordering food before arriving at a highway exit. In these scenarios, the network protocol would have to facilitate reliable, secure transactions.

A second set of applications in the highway environment is more similar to those discussed above for the home. Dumb, sensing devices unique to a highway environment might provide data about traffic conditions like congestion, flow patterns, weather, etc. These devices would likely only need to continually transmit a best-effort sampled data stream. These sensor devices may need to execute basic commands or may simply transmit a continuous flow of data.

- **Inhospitable environments:** There are a number of inhospitable environments that would benefit from an array of very simple devices that use a lightweight network protocol to communicate. Generally, we are talking about a large number of wireless sensors spread over an area to provide continuous feedback. The key difference between this class of applications and the others is that the devices would have to self-organize into a network and work together to communicate necessary information to points on the periphery of the network. A prototypical application is the blanketing of a battlefield with thousands of sensors. Sensors would collect and communicate reconnaissance information to the edges of the cloud where some intelligent agent would

gather, possibly process, and likely relay the information through some traditional network. To be truly successful sensing devices would have to be simple, cheap, and plentiful.

Other applications included in the inhospitable environments class of applications are conditions monitoring, disaster relief assistance, and search and rescue efforts. For example, seismic sensors could be scattered over a collapsed building and used to detect the motion of trapped survivors. Devices might even be installed into the building infrastructure during construction and used for search and rescue efforts in a collapsed building if it is destroyed by disasters like earthquakes or fires. These devices might also server to collect nominal environmental data like building air quality, temperature, etc.

3. Overview of the Pseudo-IP Protocol

The goals of Pseudo-IP are (1) to reduce the overhead and complexity of a full network layer protocol, (2) be flexible enough to interoperate with different medium access layer protocols, and (3) be flexible enough to provide network service in a variety of environments. Obviously the most common network layer protocol is IPv4¹. In considering what functionality Pseudo-IP should provide, we should examine what functions IPv4 provides. They include the following[5]:

- Packet length – 1 bytes
- Identification/Sequence number – 2 bytes
- Fragmentation/reassembly – approximately 2 bytes
- Time to live – 1 byte
- Upper layer protocol identifier – 1 byte
- Header checksum – 2 byte
- Source and destination addressing – 8 bytes
- Miscellaneous other bits – 2 bytes
- Options and variable length headers – variable

In addition to the overhead associated with the IP header, there is processing overhead required to implement protocol functionality. For example, to properly support IP, the Internet Control Message Protocol (ICMP) should be implemented. Furthermore, protocols to provide translation between medium access control layer addresses and network

¹We also will consider how IPv6 differs from IPv4 but the philosophy of IPv6 is similar enough that we can concentrate on IPv4 at this point.

layer addresses requires two resolution protocols: the Address Resolution Protocol (ARP) and the Reverse Address Resolution Protocol (RARP). And finally, all of this overhead is in addition to whatever overhead is required by the medium access control protocol. If devices implement a wireless protocol like the IEEE 802.11 wireless LAN standard, fewer Pseudo-IP functions will be required because 802.11 provides its own addressing and checksum mechanisms; uses collision avoidance; and has provisions for acknowledgments[6].

Other, simpler medium access control protocols might have to be used like Aloha[7] and slotted-Aloha[8]. Specifically then, what Pseudo-IP should provide is (1) a lightweight interface for communication among dumb or semi-intelligent devices, and (2) protocol translation between Pseudo-IP and traditional IP. Much of the routing functionality will be based on radio transmissions, i.e., the fact that these unconventional devices are usually broadcast-capable.

4. Research Challenges

The goal of Pseudo-IP is to provide basic network layer functionality while still allowing higher layer protocols to provide services like reliability, congestion control, authentication, etc. Dumb devices should only have to implement the minimum number of functions to achieve connectivity. Furthermore, simple devices should not incur overhead penalties for functions they cannot or do not wish to perform. Our research plan is based on creating a lightweight network layer protocol. Conceptually, our Pseudo-IP protocol can be compared to the relationship between UDP and TCP. UDP, when compared to TCP, is a lightweight protocol providing almost no transport layer services.

Pseudo-IP will eliminate most of the fields of both IPv4 and IPv6. We will investigate the issues raised by having no addressing, no routing information, no fragmentation/reassembly function, no error detection, and no sequence numbers. The basic paradigm for this simplest case will be random broadcast of data. Packets will still have a Medium Access Control (MAC) protocol which will provide framing, and probably some form of identification and basic error detection.

The research challenges we plan to explore are associated with the issues raised by providing communication using Pseudo-IP. Part of this challenge includes investigating how to build additional network services, like reliability for control functions, on top of Pseudo-IP. A second challenge is how to interconnect Pseudo-IP clouds with the existing IP infrastructure. A brief description of some specific research issues

include the following:

- **Data Flow.** Straightforward data flow should likely only require the simplest version of Pseudo-IP. For example, in the case of simple sensors, data will flow one way, not even requiring return information or feedback to the sensor. Sensors will periodically broadcast their sensor information and not care if it is ever received. These devices should be very cheap compared to an IP-capable device. Given the potential environments, there are two types of network topologies that these types of devices might have to communicate in. The first topology would not require any routing because all devices can communicate directly with the desired receiver. Devices either have a wired connection to the receiver or operate in a wireless environment where the receiver is known to be in range. A medium access control protocol would be responsible for implementing collision avoidance functionality.

The second topology assumes that all data should be delivered to a single remote receiver and not all transmitters are within range of the receiver. This type of topology requires basic routing functionality and represents a significant jump in complexity. The additional complexity includes the following components:

- How to do routing? Given that devices may be expected to perform in inhospitable environments, the network topology may actually change frequently. Furthermore, running a complex route discovery protocol is unlikely to be feasible given the nature of the devices. Our preliminary assertion is that some sort of optimized random routing or intelligent flooding algorithm should be used. A second consideration associated with broadcast-based routing is the need to remove old packets from the network. IP uses a monotonically decreasing time-to-live (TTL) field that causes a packet to be discarded when the TTL value reaches 0.
- Addressing can range in importance from critical to not necessary. For some applications, like a blanket of sensors dropped in an inhospitable environment, the actual location of a device may need to be known. Sensors may need a GPS-based locating system.
- In some cases, strict timing information will be required. Time stamps might have to be taken and

then used as sequence numbers. The medium access control protocol might provide some part of this function, for example through a hardware address or through a fully pre-configured arrangement like Time or Frequency Division Multiplexing (T/FDM).

- **Semi-Reliable Feedback.** One potential problem with not having a way to transmit feedback to a large set of dumb sensors is the total lack of control a management station would have. The problem occurs as more and more sensors are brought on-line and/or when using more capable devices (e.g., mobile sensing devices). The period between each sensor's transmissions may be too short and a large number of collisions may result reducing the effective data rate. A control station might want to communicate with the array of devices using some very simple feedback channel. In this example, the control station might select a specific interval to broadcast to all sensors. These semi-reliable control functions can be achieved by again using a broadcast paradigm. By repeating the broadcast multiple times, all or most of the sensors will eventually receive the control information.
- **Reliable Transactions.** There will likely be intelligent enough devices that will want to exchange information reliably. For instance, a host might want to reliably control a light switch, or other simple device in a room. In order to do so, some information needs to be passed between the host and the device. In IP-based communication, this is accomplished using a series of messages. In Pseudo-IP, this could be accomplished by exploiting MAC layer mechanisms such as TDMA slots and MAC-level ACKs/NACKs. While this breaks the traditional model of layered network protocol design, it greatly enhances the ability to accommodate devices that are only capable of sending small messages. The idea is to perform authentication, sequencing and reliability based on lower-layer mechanisms, rather than relying on extra network layer bits.
- **System Control.** Control functions are built on top of reliable transactions but the additional challenge is determining what parameters are available for control and identifying a way of communicating the control function. Ideally, we would like not to have to define a standard for information exchange (e.g., identifying that information is coming from a light switch and not the dish washer). Standards are susceptible to politics, inefficient

cies due to aging, and additional problems of interacting with devices that do not support the standard or may support different versions of it.

- **Security.** Reliable transactions require secure channels. End-to-end encryption can be used since devices engaging in transactions are likely to be more powerful in terms of processing and communication capabilities. When dealing with the more simple devices, physical security may be the only option. In the home environment, for instance, devices could rely on line-of-sight communication, requiring physical proximity to the devices. Remote control could be enabled through the use of an intelligent IP gateway which could provide remote authentication services.
- **Inter-Cloud Routing.** In order to allow communication between simple devices in different clouds, edge devices would act as intelligent gateways. Although devices might not be able to address other devices directly, gateways could collect Pseudo-IP packets and encapsulate them in IP packets for remote distribution. Edge devices could have static IP addresses, while intra-cloud communication is via local dynamic addressing.
- **Directory Services.** The problem of discovering local services in an area becomes problematic as the number of devices grows. Since bandwidth is limited, there must be some way to gather information without consuming all the bandwidth with service announcements. This can be accomplished by providing directory servers in a region. Although this should not be a requirement of the system (devices should still disseminate their status periodically) they can provide an optimization when improved network performance is required. Directory server would collect data on the local region and provide this information upon request. For example, and vehicle entering a cloud could request information on available services from a directory of services in the area, rather than having to wait for all services to announce their availability. If such a directory server were not available, the vehicle would have to wait longer to obtain such a list, but it could still be obtained.

One problem in designing a protocol like Pseudo-IP is the uncertainty in knowing what specifications the devices meant to use Pseudo-IP will actually have. There are questions about processing capability, bandwidth, transmission range, storage capacity, duration of operation, etc. In addition to

device specifications, there are questions about the environments these devices will have to operate. There are also questions about the type and size of data collected, the real-time requirements of data delivery, number of devices in a region, environmental hazards, etc. Our proposed research agenda will focus on designing a number of detailed scenarios and then specifying a protocol to most efficiently and effectively address the network needs.

References

1. S. Deering and R. Hinden, "Internet protocol, version 6 (IPv6) specification." Internet Request for Comments RFC 1883.
2. Daedalus Project Team, "The daedalus project home page." <http://daedalus.cs.berkeley.edu/index.html>, January 1996.
3. T. D. Hodes and R. H. Katz, "Composable ad-hoc location-based services for heterogeneous mobile clients." Submitted for review, *Wireless Networks Journal* special issue; also available from <http://daedalus.cs.berkeley.edu/publications/services-WINET.ps.gz>, November 1997.
4. J. Nonnenmacher, E. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," in *ACM Sigcomm 97*, (Canne, FRANCE), August 1997.
5. A. Tanenbaum, *Computer Networks, 3rd Edition*. Upper Saddle River, New Jersey: Prentice Hall, Inc., 1996.
6. V. Hayes, "IEEE standard for wireless LAN medium access control (MAC) and physical layer (PHY) specifications," Tech. Rep. IEEE 802.11-1997, Draft 6.1, IEEE Computer/Local & Metropolitan Area Networks Group, June 1997.
7. N. Abramson, "Development of the ALOHANET," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 119-123, March 1985.
8. L. Roberts, "Extensions of packet communication technology to a hand held personal terminal," *Proceedings of Spring Joint Computer Conference, AFIPS*, pp. 295-298, 1972.

Beyond audio-based speech recognition for natural human computer interaction

Sankar Basu, E E Jan, Mark Lucente, Chalapathy Neti

IBM T. J. Watson Research Center, Yorktown Heights. NY 10598

I. Abstract

The development of more natural and intuitive man-machine interfaces that do not require humans to acquire specialized and esoteric training such as the use of keyboards and cumbersome body-worn microphones still remains a goal. In this paper, we will outline the technical challenges and some solutions to the problem of making speech a natural human computer interface in smart spaces. We explore the combination of visual expression with audio-based expression of speech for recognition of speech and intent. In particular, we postulate the benefits of combining multiple sensory realizations for robust recognition of a perceptual process.

II. INTRODUCTION

Imagine sitting in front of a computer, untethered and browsing the web using spoken language or sitting comfortably on your couch in the family room and interacting with the television using spoken language. The development of more natural and intuitive man-machine interfaces that do not require humans to acquire specialized and esoteric training such as the use of keyboards still remains a goal. Current human-machine interfaces include the keyboards, mouse, touch screens, pens and close-talking microphones. To facilitate a more fluid interface we would like to enable machines to acquire human-like skills such as recognize facial gestures together with speech to make machine recognition and understanding of human speech more robust in natural untethered settings.

It is expected that important benefits will come from natural spoken language interfaces that provide an integration with speaker independent speech recognition, natural language understanding and natural sounding speech synthesis. However, a critical role will be played by robust ways to process, and indeed to emulate, nonverbal human expressions such as the facial (and perhaps also manual) gesticulation. We naturally use our faces as organs of expression, reaction, query and also to modulate visually or to add nuance to speech. Facial expressions, including head movements, are often used in lieu of and in combination with speech to express assent, dissent, uncertainty, irony and other reactive states in human communication.

All of these additional facial features and emotional states can be important contributors to the robust recognition of speech and understanding the intent of human action.

We have made significant progress in speech recognition for well-defined applications like dictation and medium-vocabulary transaction processing tasks in relatively controlled environments. However, for speech to be a pervasive UI in smart spaces in the same league as graphical user-interfaces is for desktops, it is necessary to remove some impediments that exist today. These include:

- lack of robustness to changes in environment, sound-channel and speaker
- lack of hands-free/tether free input making speech applications less natural and cumbersome. Inability to handle far-field input with the same fidelity as a close-talking microphone

In this paper, we suggest and motivate the combination of visual expression features with audio-based speech expression for robust recognition of speech in smart spaces. More specifically, we will consider the following:

- combination of visual and acoustic realizations of the speech stream for robust speech recognition by using hands-free/tether free far-field acoustic input
- use of speech-source localization to augment image-based extraction of facial features relevant to speech recognition and to augment far-field sound capture.
- combine other aspects of the human communication embodied in the facial features to augment the understanding of the intent of a speech utterance.

III. VISUAL CUES FOR SPEECH RECOGNITION

At present the most effective approach for achieving robustness of environment focuses on obtaining a clean signal through a head-mounted or hand-held directional microphone. However, this is neither tether-free nor hands-free. Moving the speech source away from the microphone can degrade the speech recognition performance due to the contamination of the speech signal by other extraneous

sound sources. For example, using omnidirectional monitor microphones for far-field input can severely degrade performance in the presence of noise, but on the other hand using directional desktop microphones constrain the extent of movement of the speaker making the interaction unnatural.

Speech recognition performance today is very sensitive to variations in the channel (desktop microphone, telephone handset, speakerphone, cellular, etc.), environment (non-stationary noise sources such as speech babble, reverberation in closed spaces such as a car, multi-speaker environments, etc.), and style of speech (whispered, Lombard speech, etc.). Several signal-based and model-based techniques to make speech recognition independent of channel and environment have been attempted with limited success [9, 7]. Most techniques make strict assumption on the characteristics of the environment and require a sizable sample of the environment to get small improvements in speech recognition performance. Furthermore, modeling reverberation is a hard problem. In summary, current techniques are not designed to work well in severely degraded conditions.

We need novel, nontraditional approaches that use other orthogonal sources of information to the acoustic input that not only significantly improve the performance in severely degraded conditions, but also are independent to the type of noise and reverberation. Visual speech is one such source not perturbed (obviously) by acoustic environment and noise. Canonical mouth shapes that accompany speech utterances have been categorized, and are known as visual phonemes or "visemes" [16]. Visemes provide information that complements the phonetic stream from the point of view of confusibility. For example, "mi" and "ni" which are confusable acoustically, especially in noise situations, are easy to distinguish visually: in "mi" lips close at onset, whereas in "ni" they do not. The unvoiced fricatives "f" and "s" which are difficult to recognize acoustically belong to two different viseme groups.

Current speech recognition systems do not use any visual information to augment the audio signal. Some reasons that motivate the use of visual input are:

- Human recognition improves from 23% accuracy in severely degraded conditions (less than 0 dB SNR with interfering speech noise) to 65% by using visual information in addition to the acoustic signal [17]. Preliminary automatic speech recognition experiments using visual speech reduce the error rate by about 30-50% on small speech tasks but under cross-talk conditions at 15dB SNR [3, 12].
- Video input is becoming cheaper and prevalent as audio input in multi-media computers today. CCD based camera's for PCs with reasonable resolutions are available for well under 100 dollars today.
- Software based real time video/image feature extraction (such as lip-tracking) can be implemented on Pentium-class personal computers [14].

A. Psychophysics of visual speech recognition

To elaborate on the motivation for the use of visual input for speech recognition, we describe next a well known psychophysical experiment (Summerfeld, '79). In this experiment, acoustic signal generated by a person reading 30 sentences was fused with another equally loud acoustic signal (considered interfering noise) generated by a second human reading arbitrary prose. Ten human transcribers were asked to transcribe the speech under the following conditions: (A) Acoustic Only; (B) Acoustic + full video; (C) Acoustic + lip region; (D) Acoustic + four points on lips. The transcription word accuracies for Condition (A) was the worst at 23%, while for Condition (B) it was the best at 65%. For Condition (C) and Condition (D), the transcription accuracies were 54% and 31%, respectively. Better transcription rate was obtained when video of the entire face was made available rather than the condition when only the lip region was displayed. This suggests that humans indeed extract more facial features than the region around the lips to perform visual recognition. However, exactly what features are extracted by humans remains unclear.

B. Visual features for facial expression

Although previous work has been conducted to define the viseme units derived from human lip-reading experiments [15] and other psychophysical data, more research is necessary to identify the mouth features that are relevant for large vocabulary, speaker independent visual speech recognition. Some possible features are gray-scale parameters of the mouth region; geometric/model based parameters such as area, height, width of mouth region; lip contours arrived at by curve fitting, spline parameters of inner/outer contour; and motion parameters obtained by 3-D tracking. Gray scale parameters suffer from being sensitive to lighting conditions. Lip contour information, although invariant to lighting conditions, may not provide enough information of the inner articulators such as teeth and tongue. Thus, it is necessary to investigate an optimal combination that is appropriate for visual speech recognition.

Similarly, a set of forty-four dynamic features known as the action units (AU) have been defined and has been used by psychophysicists for describing facial expression [1]. However, a parametric description of these AUs that are usable by machine recognition algorithms for facial expression has not been adequately defined. Among several suggestions, one pursued by image coding community is to describe the face by a wire-frame type model. The control points of the wire-frame model then provide the paramet-

ric description of the AUs just mentioned. Other more sophisticated models that possibly include the time dynamics of the parameters can be envisioned. Exploration of other parametric representations for facial expression for speech reading can be proposed as a research agenda for the immediate future.

To elaborate on the actions of facial gesture, it needs to be mentioned that the action involves both temporal and spatial components. The temporal aspect demands that the gesture model we use be time scale invariant. For example, the action of eye blinking should be recognized as such, independently of whether it is slow or quick. Recognition scheme should also be independent of the variability of the parameters from person to person.

In speech recognition Hidden Markov Models (HMM) have shown tremendous success in modeling spoken words or other speech units independent of their variation in duration and in pronunciation. An HMM is a doubly stochastic process with a hidden state process that drives the model dynamics and an output process that is usually modeled by a mixture of Gaussian densities. As in automatic speech recognition, we envision associating an HMM with each gestural action. Parameters of these gestural action units (AU) can be those described earlier. The HMM parameters for such models are then to be estimated from data via the process of ‘training’ from large amount of data collected from labeled gestures, that maximizes the likelihood of observing the data. The problem of recognition is to evaluate the probability of the gestural actions given the HMM model parameters and to pick the most likely gesture.

The situation is highly analogous to modeling words or sub-word units (e.g., phonemes) in speech, and there exists a wealth of well proven techniques to achieve this goal. Clearly, use of a larger set of gestural units is likely to recognize the gestural action more accurately, but at the expense of greater amount of computation. This problem can be alleviated by appealing to the fact that all sequences of gestural actions are not equally likely to follow in time due to restrictions imposed by psychophysics (e.g., facial muscles may not allow certain actions followed by others, whereas expression of sadness is unlikely to immediately follow wide laughter). Modeling of such external psychophysical constraints can also be carried out in statistical terms via the collection of large amount of data. Thus, one envisions a ‘gestural grammar’, the modeling of which can in principle, be performed either by ‘rule based’ or statistical techniques. Again, the situation is analogous to the use of ‘Language Models’ in speech to aid the acoustics only based recognition. It may be worthwhile to note here that although rule based techniques have been investigated in the past for such purposes, statistical methods are currently favored in the linguistic as well as in the speech recognition community [20].

IV. MICROPHONE ARRAYS FOR MULTI-MODAL SPEECH

For man-machine communication via speech to be natural one needs hands-free sound capture with no tether or encumbrance by hand-held or body-worn sound equipment. However, performance of sound pickup via a single microphone is degraded prominently by deleterious properties of the acoustic environment, such as ambient noise and multi-path distortion, when the microphone is positioned far away from the acoustic source.

In a multipath environment, the output of a transducer, $x(t)$, associated with the far-field point source signal, $s(t)$, can be simplified as:

$$x(t) = \sum_{i=0}^k \alpha_i s(t - \tau_i) + n(t) \quad (1)$$

where k is the number of reflections considered, α_i is the attenuation with respect to the i^{th} reflection, τ_i is the corresponding delay, and $n(t)$ is the ambient noise. The direct path is presented when $i = 0$.

To design a single microphone with super-directive pattern to recover the original signal in this multipath environment is a great technical challenge. Instead, several approaches using multiple microphones have been proposed to discriminate the reverberation and attenuate ambient noise. One technique is inverse filtering where an inverse filter of room impulse response is designed. This approach is effective but fragile. The filter is not always causal and is very sensitive to environment changes. A slight change in room temperature can severely degrade the performance. Another approach is adaptive beamforming by using Minimum Mean Square Estimation (MMSE) to eliminate the unwanted signals. Since reverberations are replica of the source signal with delays, they can not be effectively eliminated by MMSE. Design of better criteria is key to improve the performance in a highly reverberant environment.

An attractive approach is the delay-and-sum type beamformer. This requires the knowledge of source location to calculate the delay between each microphone. This method is simple and effective in a moderated reverberant environment. However, in a severe multipath environment, the beam also picks up all of the reflected and delayed signals (images) along the beam direction. More sophisticated array processing techniques are necessary to improve the sound capture quality in severely degraded environments [6].

Three main issues needed to be addressed for delay-and-sum type beamforming are source localization, sound capture and array configuration. Source localization algorithm estimates the location of active acoustic source using a time delay estimator [2]. The performance of time delay estimator on wide-band speech signals using cross correlation in either time or frequency domain is

degraded in the multipath environment. A better technique is required to improve the source localization accuracy [4]. Different array signal processing techniques are then applied to process each channel using the given source location information to provide better audio output. In addition, such localization can also assist camera steering/focusing in a multi-modal environment. A better array configuration that distributes microphone more effectively may improve source localization accuracy and sound capture quality, with fewer transducers and computation resources. A randomly distributed array is an attractive approach. [5].

Microphone arrays have been applied to speech recognition in deleterious environments. The following is a simple digit recognition task using small scale microphone array (10-20 microphone elements) as audio input device. In the digit recognition task with the speech source 1.5 meter away from the transducer, the array provides a recognition accuracy of 74.3% while an omnidirectional microphone provides 52.5% accuracy. The accuracy with a close talking microphone is 99.5% [13]. The accuracy of small-scale array suffers relative to a close-talking microphone due to the loss of amplitude of speech signal (leading to a low SNR) and a mismatch between the estimated and actual direction of arriving signals.

The current microphone array based speech recognition can be greatly enhanced by the use of speech-source localization and augmentation of the audio signal by visual cues. Knowing the precise location of the speech source can help selectively enhance the speech signal. Speech source localization can help to locate the facial features required for visual speech recognition. Use of visual cues can help improve recognition in a low SNR condition.

As an example, we consider IBM "DreamSpace" scenario [18] where the system "hears" the human users and "sees" their gestures and body positions, allowing users to collaborate in a shared workspace via audio-visual computing. One of the limitations of this concept is the fact that the user has to wear a wireless microphone to get reasonable speech recognition performance. Such a multi-modal interaction using gestures and speech can become truly natural if we provide hands-free/tether-free speech input and robust speech recognition via the simultaneous use of audio based localization of the active speaker and visual cues. In addition, the "Dreamspace" demo is susceptible to visual clutter if too many people are present. At present, a multi-class statistical model of color and shape is used to segment the profile of the object (active human) of interest from the background. Such schemes for identifying the human "actor" can become more effective if augmented with speech based localization of the speaker.

V. MODE FUSION FOR MULTI-MODAL SPEECH RECOGNITION

A. Data fusion aspect

Using visual features to augment the audio signal for speech recognition involves the ability to fuse different realizations of the same underlying perceptual process. Such a mode-fusion or multi-modal integration involves the following three categories of sensory data fusion [11, 8].

- data fusion — this involves integration of different modalities in raw form e.g., video camera and microphone outputs.
- feature fusion — features are extracted from the raw data and subsequently combined. This involves e.g., speech features with lip and facial features such as the action units (AU).
- decision fusion — this is the fusion at the most advanced stage of processing and can happen at the sub-word level, word-level, utterance level or at the action level.

In general, progression from data fusion to decision fusion provides a higher degree of robustness, but is accompanied by possible loss of information. An optimal fusion policy of using one of these fusion strategies or some weighted combination of the three strategies needs to be investigated.

B. Compositional aspect

While the above considerations have to do with fusing together different forms of data, a hierarchical representation, often exploited in modeling of perceptual organization and one that is currently being intensely investigated in image processing [19], would be useful in our problem. In this representation, models to be detected are decomposed recursively into smaller units, leading to a hierarchy of models. Computation is mostly a bottom up procedure which detects and reconstructs these models by composition. Top down feedback, whose laws are derived from model hierarchy is used to select groupings of primitive units which are consistent with high level models. A hierarchy of descriptions obtained by recursive composition of descriptors is generated from this process.

Furthermore, the desirability of a smooth hierarchy demands that the adjoining levels in the hierarchy be sufficiently close to each other so that every level contains enough information to construct the models at the following level. Such hierarchical representation can be exemplified by considering the decomposition of speech into sentences, words, syllables, phones, and finally into further sub-phonetic units. Visual or other cues of the same perceptual process can be added at appropriate levels of this hierarchy as well.

Using this *compositional* approach it is possible to resolve uncertainties in recognition *only when* adequate contextual information is available. Since often it is not possible to eliminate all uncertainties in a single step, one eliminates part of the uncertainty first, and then uses new, more informative representation at a higher level to resolve further uncertainty. This *deferred decision making* until higher levels in the hierarchy is reached is the key to the *compositional approach*. In general, intermediate representations may be quite uncertain and ambiguous and may even allow mutually inconsistent hypothesis.

VI. SOME APPLICATION SCENARIOS OF MULTI-MODAL SPEECH RECOGNITION

A. Detection of intent to speak

As a simple illustration of the application of sensor fusion, consider the problem of robust detection of the start of a speech event in natural spoken language interaction. We follow the push to talk paradigm today using a mouse or keyboard interface. A possible hands-free solution to the problem is to first detect the event when the user faces the computer, then turn the microphone on and subsequently correlate the visual activity and audio activity to confirm the "microphone on" decision. Using the audio and visual sources to detect the onset of speech activity can be more robust than either of the channels in isolation, because it is based on correlation of two different realizations of the same speech activity. We assume the camera is constantly "seeing".

B. Spoken language transaction processing

In applications such as speech based transaction processing in a desktop environment or spoken-language interaction with television the user is forced to watch the display for output information. Thus, the mouth position of the user can be relatively well localized to a small region in front of the display. This makes the process of visual feature extraction easier compared to the general face-localization and feature extraction problem. Combining visual features of speech and auditory features can be beneficial for such applications. Having a hands-free/tether-free speech input will make the interaction more natural.

C. Multi-media gisting for video cataloging and searching

Several commercial products exist today that allow cataloging of videos based on intelligently slicing the video into logical segments based on scene change detection and speech/music segmentation. Other image based techniques such as face-detection and motion flow estimation are emerging as useful methods to classify scenes. Video classification based on transcription of the audio stream and text-based information retrieval techniques is becoming yet another useful scheme. Current state of the art (driven by ARPA Broadcast News transcription task) in

transcription accuracies can be further improved by usage of visual cues. For instance, the transcription accuracy for a field reporter reporting from a noisy environment is very poor today. Using visual cues to augment acoustic speech recognition is viable since the face of the reporter is relatively stationary in front of the microphone.

D. Smart cars

Automobiles are shaping to be smart spaces of their own. Automobiles of the future will allow for automatic temperature and music adjustment to your liking upon entry. Conversation with a virtual assistant to get important messages, news items, traffic and navigation information can also be envisioned for the near future. Several technologies have to be improved and enhanced to realize the above scenario. Improved speech recognition, multi-modal user authentication (based on body-sensing, face identification, etc) and high-bandwidth voice/data connections. Background noise and reverberation are major impediments to large-vocabulary speech recognition applications today in such environments. In such applications, the driver position is relatively fixed. Localization of the speech source to improve the fidelity of sound capture and combination of visual speech with auditory speech for improved speech recognition becomes a potentially viable approach.

VII. IMPACT ON OTHER TECHNOLOGIES

In addition to speech recognition, the same problems of channel and environment dependence arises in speaker identification. Again, the problem can be alleviated by combining visual signatures of the speaker both in terms of characteristics of visual speech and other facial features to perform speaker identification. Several sites have begun efforts in using visual speech characteristics as signatures for speaker recognition (e.g., IDIAP, Switzerland). While face recognition is a notoriously hard computer vision problem, and speech alone has seen only limited success, a combination of speech with facial cues can be beneficial in increasing the accuracy of the present systems based on either audio-only or video-only modalities.

While the specific ideas proposed in this paper are an outgrowth of speech technology, impact on other areas within the current and future activities can be foreseen as well. These include efforts in multimedia networking and authoring; ultra-high compression technologies where picture information is coded solely as text for subsequent transmission; integration with natural language interface where machine participates in negotiation and requests clarification using visual as well as auditory cues;

VIII. CONCLUSIONS

For speech to be a pervasive UI in the converged world of computing, entertainment and communication, it is necessary to bring together expertise in various technol-

ogy efforts such as speech recognition, computer vision, psychophysics, etc. For building truly natural human-computer interfaces, we believe it is necessary to begin combining the various communication modalities such as speech, gesture (facial and hand) to create a compelling and natural interaction. The research community, at large, currently has significant technology strength in each of these areas (speech, computer vision, etc.), independently. However, a small effort exists in exploiting the synergy in these diverse (but related) technologies for a true multi-modal human computer interface (eg. [21]). Simultaneous use and interaction of different technologies (speech and video) by fusion of different realizations of the same perceptual process, proposed in this paper can build on the current speech recognition technology and the face localization and understanding technology developed in exploratory computer vision.

REFERENCES

- [1] Hiroshi Harashima, Kiyoharu Aizawa and Takahiro Saito, "Model based analysis synthesis coding of video telephone images - conception and basic study of intelligent image coding", Transactions of IEICE, vol. E 72, no. 5, May 1989.
- [2] M. Brandstein, H. Silverman, "A practical methodology for speech source localization with microphone arrays", Computer, Speech and Language, vol 11, pp. 91-126, April, 1997.
- [3] C. Bregler and Y. Konig, "Eigenlips for robust speech recognition", ICSLP, vol II, pp. 669-672, 1994.
- [4] E. E. Jan and J. L. Flanagan, "Sound source localization in reverberant environments using an outlier elimination algorithm", Proc. ICSLP, pp. 1321-1324, Oct. 1996
- [5] E. E. Jan and J. L. Flanagan, "Sound capture from spatial volumes: matched-filter processing of microphone arrays having randomly-distributed sensors", Proc. ICASSP, pp. 917-920, May 1996
- [6] J. L. Flanagan and E. E. Jan, "Sound capture with three-dimensional selectivity", Acoustica 83(4), pp. 644-652, 1997.
- [7] Mark Gales, "'Nice' model based compensation schemes for robust speech recognition", ESCA-NATO workshop on robust speech recognition for unknown communication channels. Pont-a-Mousson, France, pp. 55-59, April 1997.
- [8] David L. Hall, "Mathematical Techniques in multi-sensor data fusion", Artech House, 1992.
- [9] B. H. Juang, "Speech recognition in adverse environments", Computer Speech and Language, vol 5, pp. 275-294. 1991.
- [10] Casper Horne (ed.), "SNHC Verification Model 4.0" ISO, coding of moving pictures and audio, ISO/IEC/JTC1/SC29/WG11, Document no. N1666, MPEG 97, April 97.
- [11] R. Sharma, V. I. Pavlovich and T. S. Huang "Towards multi-modal human computer interface", Proc. IEEE, Tsuhan Chen, K. J. Ray Liu and Murat Tekalp ed., pp. 853-869, May-June 1998.
- [12] G. Potamianos and H. P. Graf, "Discriminative training of HMM stream exponents for audio-visual speech recognition", ICASSP, 1998.
- [13] E. Leida, J. Fernandez, E. Masgrau, "Robust continuous speech recognition system based on microphone array", Proc. ICASSP. Seattle, WA, 1998.
- [14] R. Stiefelhagen, U. Meier and J. Yang, "Real-time lip-tracking for lipreading", preprint.
- [15] D. Stork and M. Hencke, "Speechreading by humans and machines", NATO ASI Series, Series F, Computer and System Sciences, vol.150, Springer Verlag, 1996.
- [16] D. W. Massaro and D. G. Stork, "Speech Recognition and Sensory Integration", American Scientist, pp. 236-244, May-June 1998.
- [17] Q. Summerfeld, "Use of visual information for phonetic perception", Phonetica(36), pp. 314-331 1979.
- [18] Mark Lucente, "Visualization Space: A Testbed for Deviceless Multimodal User Interface ", Int. Env. Symp. AAAI, Stanford, March 1998.
- [19] Sanjoy K. Mitter, Personal Communication. Also, Stuart Geman, "Three lectures on image understanding", The Center For Imaging Science, Washington State University, video tape, Sept. 10-12, 1997.
- [20] Eugene Charniak, "Statistical Language Learning", Language Speech and Communications Series, MIT Press, 1996.
- [21] First IEEE workshop on multimedia signal processing, June 23-25 Princeton, New Jersey, 1997.

AutoAuditorium.TM a Fully Automatic, Multi-Camera System to Televis Auditorium Presentations

Michael H. Bianchi

Bellcore Applied Research
Morristown, NJ 07960

ABSTRACT

A large room full of people watching a presentation suggests that there are other people, unavailable at that time or not at that location, who would like to see the talk but can not. Televising that talk, via broadcast or recording, could serve those absent people.

Bellcore's AutoAuditorium¹ System is a practical application of a Smart Space, turning an ordinary auditorium into one that can automatically make broadcasts and recordings. The system is permanently installed in the room and uses optical and acoustic sensors (television cameras and microphones) to be "aware" of what is happening in the room. It uses this awareness to televise the sound and images of the most common form of auditorium talk, a single person on a stage, speaking with projected visual aids to a local audience.

Once turned on, the system is completely automatic. The person on stage and the people in the local audience may not even be aware that it is on. To remote audiences, the program is usually as watchable as one produced by a one-person crew running the system by hand.

This paper describes the system, some of our experiences using it, and planned enhancements and research.

1. AUTOAUDITORIUM SYSTEM DESCRIPTION

The prototype AutoAuditorium System is installed in the largest meeting room at Bellcore's Morristown New Jersey location. The system consists of a computer with two video frame grabbers, three fixed cameras (pointed at the stage, the screen, and the lectern from the side), one tracking camera under computer control that follows the person on the stage, a video mixer, also under computer control, and several automatic audio mixers. The system is organized into three main subsystems.

The AutoAuditorium Tracking Camera follows a person on the stage, panning, tilting, zooming and focusing in response to her movements.

The AutoAuditorium Director controls the video mixer, selecting among the four cameras and a combination shot (slide screen + presenter) using heuristics that produce quite watchable programs from most presentations.

The AutoAuditorium Sound mixes sound from an optional wireless microphone, microphones installed above the stage, and microphones installed above the audience seating area. The stage microphones provide adequate audio coverage if the wireless microphone is not used or fails, and they also feed the room's public address

system. The Sound subsystem gives preference to voices originating from the stage, but also listens for audience questions.

The outputs of these subsystems create a television program that is then distributed via various mechanisms, video cassette recording, video network, and computer-encoded recording and transmission.

In the current system, each of the subsystems operates independently, although the Director changes parameter settings in the Tracking Camera algorithm for some shot selections. We plan to add more cross-subsystem awareness.

1.1. AutoAuditorium Tracking Camera

The AutoAuditorium Tracking Camera follows the person on the stage without requiring that they wear or carry anything or that they be identified in advance to the system. (There are other tracking cameras that identify their targets via devices worn by the person, or by an operator identifying a "visual signature". Either of these techniques would have interfered with the goal of making the system totally automatic and unobtrusive.)

Instead, a "Spotting Camera", mounted close to the Tracking Camera, is pointed at the stage area and its signal goes to one of the frame grabbers in the computer. A Search Area, where the person on the stage will be walking in the Spotting Camera image, is defined during installation. A map is defined that relates points in the Spotting Camera image to pan, tilt, and zoom positions of the Tracking Camera. The Tracking Camera software detects any motion in the Search Area and drives the Tracking Camera to the appropriate pan, tilt, and zoom position. (The Search Area also keeps the seated (and sometimes standing) audience motion from becoming important to the Tracking Camera.) See Figure 1.

Several parameters are set during system installation to tune the various tracking and smoothing algorithms:

- Parameters associated with the particular brand and model of pan-and-tilt mount, lens, and camera used in the Tracking Camera subsystem.
- A minimum and maximum target shape and size.
- Areas where there may be "extraneous" motion.
- Parameters that affect the responsiveness of the tracking algorithm

An example of an extraneous motion occurs when the projection screen is within the Search Area that a person may occupy. Defining the portion of the Spotting Camera image that is the projection screen as an Extraneous Motion Area helps the algorithm discriminate between motion due to the person and motion due to the visuals

¹AutoAuditorium is a trademark of Bellcore.

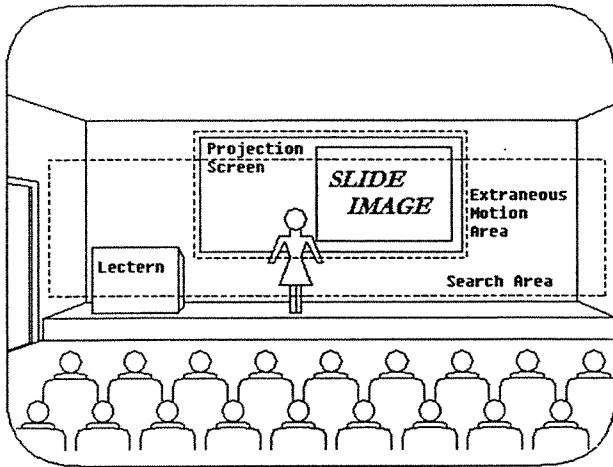


Figure 1: Spotting Camera Image, with Search Area and Extraneous Motion Area Defined

changing. (Figure 1) When the person is not standing near the screen, there is no confusion. Should the person be near the screen when the slide changes or animates the algorithm may see the motion on the screen and the motion of the person as related. If it does, the Tracking Camera zooms out to include both in the shot.

1.2. AutoAuditorium Director

The AutoAuditorium Director's function is to present camera shots that will be interesting to the remote audiences. It is driven by analyzing the image on the projection screen, viewed by a fixed camera called the Slide Camera. That image goes to both the video switcher and the second frame grabber in the computer.

The Director analyzes the Slide Camera image to determine if the projection screen is blank. If so, it directs the video mixer to show the Tracking Camera, following the speaker as he moves around the stage and talks to his audiences. See Figure 2.

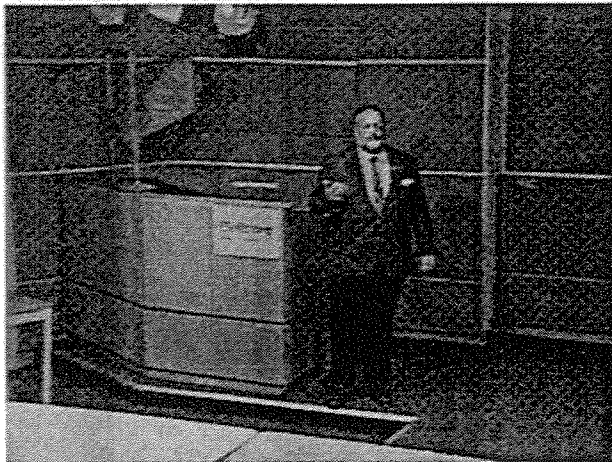


Figure 2: Tracking Camera Shot of Speaker, Alone.

Should a slide be projected, the Director sees that the Slide Camera

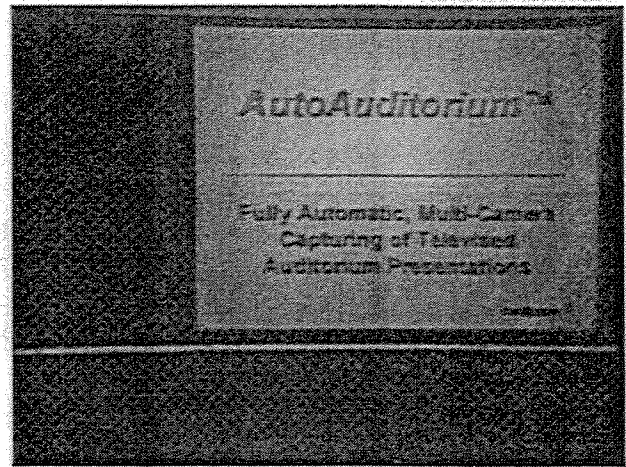


Figure 3: Slide Camera Shot of Projection Screen, Alone.

image is no longer blank and quickly directs the video mixer to show it. See Figure 3.

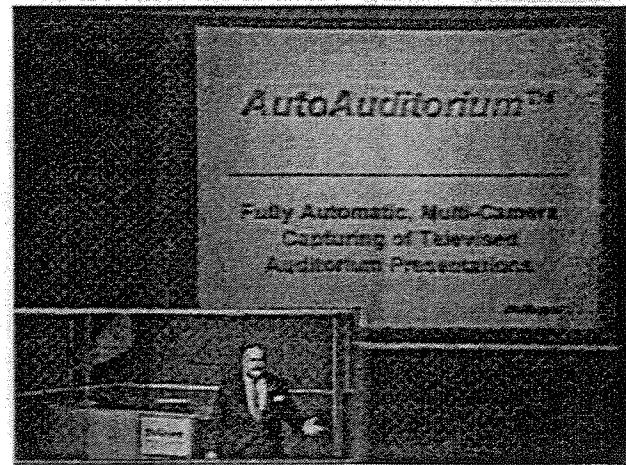


Figure 4: Combination Shot: Slide Camera with Tracking Camera Picture-In-Picture

Since it is not yet possible to determine automatically whether the most important image should be of the speaker or of the screen, a "combination shot" is constructed, with the speaker placed in a picture-in-picture box in the lower corner of Slide Camera image. See Figure 4. The picture-in-picture appears after a brief delay, since the Tracking Camera algorithm needs time to adjust to the new parameters that the Director sends it.

If the screen goes blank (Figure 5), or if the slide is unchanging for a long time, then the Director selects a "covering shot" (Figure 6) from one of the other two fixed cameras, while the Tracking Camera algorithm is reset to track the person in the center of the image. Then the covering shot is replaced with the Tracking Camera shot, Figure 2.

Should there be motion on the projection screen, or should the slide remain unchanged for an even longer time, the Director then recon-

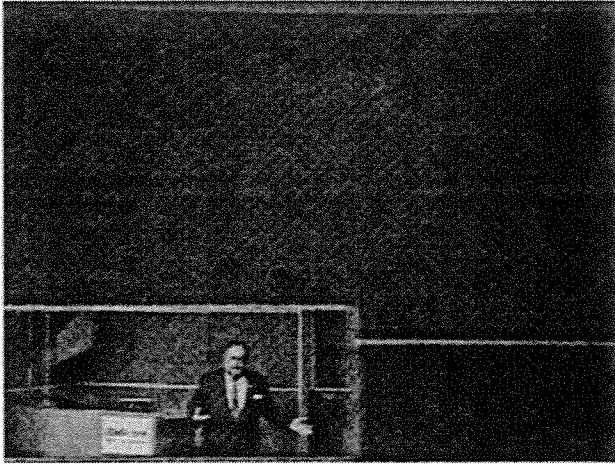


Figure 5: Combination Shot: Blank Projection Screen

structs the combination shot. See Figure 7. Because the slide image is quickly recalled to the program if there is motion within it, the Director often selects that shot just as the speaker is making a point about, and pointing at, the slide.

Simple, But Effective This simple heuristic, determining whether the projection screen is blank or not, has proved surprisingly effective in creating watchable programs of auditorium talks. Most of the time, the image on the screen is one with which the remote audiences can identify. A slide screen that has not changed in a long time (90 seconds in the Morristown installation) is generally not missed. Bringing the slide image back periodically lets the remote audiences refresh their memories about slide's content.

1.3. AutoAuditorium Sound

The AutoAuditorium Sound subsystem listens to sound from the stage, sound from the audience, and sound associated with presentation projectors. It produces a final mix from these sources.

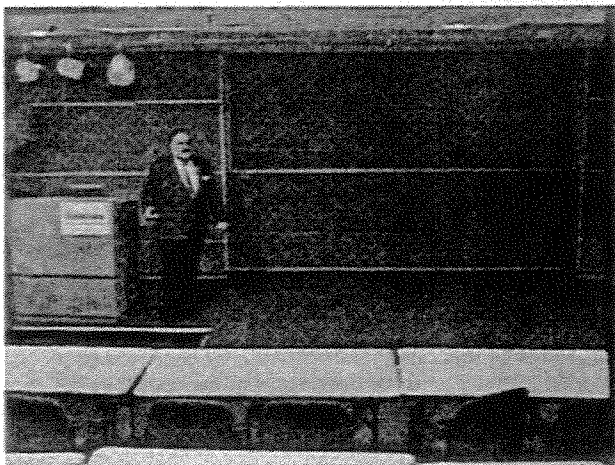


Figure 6: Covering Shot

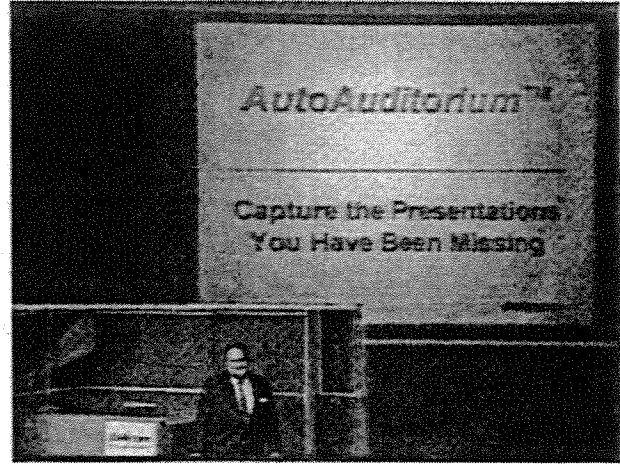


Figure 7: Back to the Combination Shot

Stage Sound In an ordinary auditorium, it is not uncommon to require that the speaker either stand at a lectern's microphone, or stand in front of a microphone stand, or wear a wireless microphone. But in a modest size room, with seating for 100 or fewer, sound system amplification for the speaker may not be strictly necessary for the local audience. Still, some form of audio pickup is required for the remote audiences.

In the Morristown Auditorium, the ceiling over the stage is low enough that six microphones, carefully placed, provide adequate audio coverage of anyone standing on or near the stage. An automatic microphone mixer combines them with the signal from the wireless microphone receiver and a microphone built into the lectern. It is so effective at selecting the best sound source into the program that we just leave the inputs at standard settings. The output from this mixer is used both for the room public address (PA) system and as part of the AutoAuditorium Sound feed. See Figure 8.

Audience Sound A similar system of ceiling microphones and an automatic mixer is used to cover the audience seating area, but with a crucial difference. Since the PA speakers are also on the ceiling over the audience, their sound would be heard by the audience microphones and cause a "bottom-of-the-barrel" reverberation. To prevent this, a simple circuit, referred to as the "Mic Ducker", mutes the audience microphones whenever the room PA system "speaks". This gives the sound from the stage precedence and keeps general audience rustling from being an annoying part of the AutoAuditorium program sound. However, it also allows the remote audiences to hear the reactions and questions of the local audience.

Projector Sound A third audio source is sound associated with projections, either from video tape or computers. In our Morristown auditorium, this "HiFi System" has its own amplifiers and speakers, under the projection screen. This signal is "tapped" and provided to the AutoAuditorium Sound mix.

The Final Mix The three audio feeds, from the stage, the audience, and the projectors, are mixed together by a final automatic mixer. Again, the strongest signal source or sources dominate the mix, and the master level is kept within the limits. The result is generally acceptable, although soft-spoken audience members are sometimes difficult to hear, both in the room and in the the program.

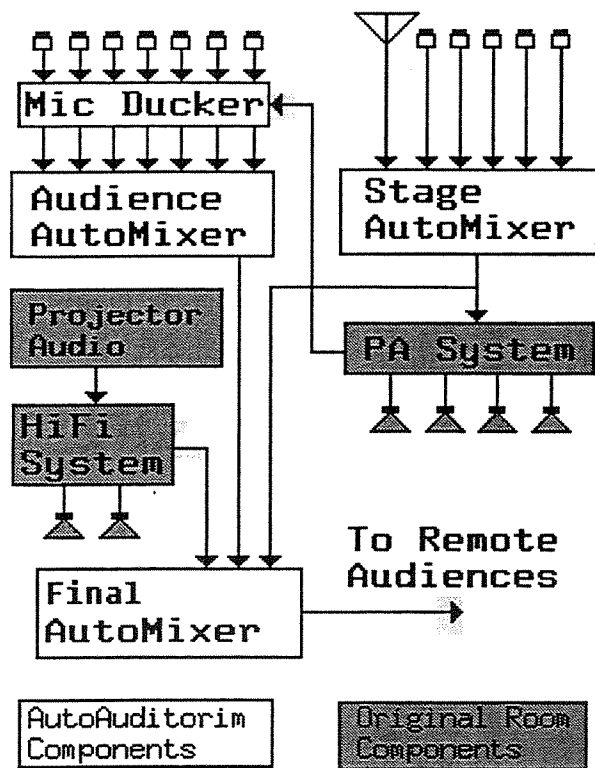


Figure 8: AutoAuditorium Sound System

2. AUTOAUDITORIUM EXPERIENCES

The idea of being able to telecast auditorium talks anywhere within Bellcore's New Jersey locations originated in the late 1980s. By the end of 1993 we had four auditoriums equipped with manually operated 3-camera systems. The expectation quickly grew that any talk of importance in any of those auditoriums *would* be telecast over our in-house T1 video network.

But the auditoriums can sometimes get very busy, with two and even three separate events in a single day. Operators stuck at the control console all day became bored and tired and would make mistakes. The operators also had other duties and were sometimes difficult to schedule.

As computer vision systems became more capable, experiments in using vision analysis to drive a tracking camera and a video mixer showed promise. By 1994, the first version of a research prototype AutoAuditorium System became operational in our Morristown NJ auditorium. Weekly work-in-progress talks were sent live over our experimental desktop video teleconferencing system, called *Cruiser/Touring Machine*[1], and also recorded for Cruiser's on-demand playback service. These weekly tests led to more refined algorithms and tuned parameters. Eventually, many people watching programs produced by the AutoAuditorium System could not tell the difference between them and manually produced programs. In fact, the AutoAuditorium programs were sometimes superior to those produced by hand because the operators would sometimes day-dream; producing a program can get very tedious.

Recently, the prototype system was ported from a locally written real-time operating system running on a single board computer in a VME card cage and using VME frame grabbers. The production system now runs on an IBM-compatible PC running Linux with PCI-bus frame grabbers.

While the system works well, it cannot fix badly prepared or presented talks. For example, visuals that can not be read easily from the back of the room are also difficult to see on television. A human operator can sometimes improve the situation by taking closeups of portions of the projection screen, illustrating the points the speaker is making. Such a capability does not yet exist in AutoAuditorium.

3. SYSTEM ENHANCEMENTS AND FUTURE RESEARCH

A number of improvements and further investigations are under consideration:

More Than One Person on Stage The Tracking Camera algorithms work well only when there is one person to be tracked. They do several things to keep from being distracted by other people momentarily crossing the Search Area, but if two people stand on the stage, the resulting Tracking Camera image is fairly unpredictable.

The production system has considerably more processing power than the prototype, so it should be possible to identify multiple people in the Search Area, especially when they are well separated. That would help the Tracking Camera to stay with the original target, or to decide to zoom out to cover both targets until one or the other left the scene.

Or, the one tracking algorithm could drive multiple tracking cameras, say with very different view points. When only one person was on stage, the ability to change camera angles could help provide variety to the program. When more than one person was on the stage, separate cameras could be assigned to separate people.

Cross-Subsystem Awareness The several subsystems of the AutoAuditorium currently run very autonomously. With the exception of the Director changing some Tracking Camera parameters as it moves the speaker's image from a full-screen shot to picture-in-picture shot, the visual processes run independently. The AutoAuditorium Sound does not connect to the computer at all. However, it is easy to enumerate benefits in making the different subsystems more aware of each other.

For one, the Director could be aware of circumstances where the Tracking Camera does not move for a long time. Some speakers place themselves behind or next to the lectern and stay there. If the Director could be aware of that, it could decide to take other shots, say of the whole front of the room or of the audience, just to provide some variety.

Another possibility, given the enhancement to track more than one person on stage, could be to use the whole-stage fixed camera shot when more than one person occupies the stage, especially if the whole-stage shot covers a wider area than the Tracking Camera can.

Multiple microphones over the stage area should make it possible to know approximately where sound is coming from. Again, given the enhancement where the Tracking Camera can identify several people on stage, that information could help the Director and/or Tracking

Camera decide which person to show to the remote audiences.

4 Pt. 2, p. 2697, October 1996 (132nd Meeting of the Acoustical Society of America, December 1996).

Seeing Audience Members The one area where a human operator can clearly outperform the AutoAuditorium product is when audience members ask questions of the person on stage. The audience microphones pick up the questions of the local audience and they are heard by the remote audiences. A human operator can point a camera into the seating area and find the person asking a question but currently the AutoAuditorium System cannot.

Rutgers University has Array Microphone technology, sometimes referred to as **Speaker Seeker**[2][3] that can stereo locate the position of a sound source. We have an early version of Speaker Seeker installed in the Morristown Auditorium, but it remains to be integrated with the AutoAuditorium system. When a person in the audience speaks, Speaker Seeker can usually point a camera at her. If that image, and the confidence measure from Speaker Seeker indicating the likelihood that it had a good image, were made available to the AutoAuditorium System, then the Director could decide to include the image of the questioner along with the sound of her voice.

Passive Micing Using Array Microphones The Rutgers Array Microphone technology could also be used to pick up the questioner's voice, instead of ceiling microphones. There may be rooms where Array Microphones on the walls could do a superior job to overhead microphones, such as when there are high ceilings over the stage and audience. We would like to investigate expanding the "nothing to wear to be heard" aspect of our current installation to more challenging spaces.

4. CONCLUSIONS

As the number and reach of high bandwidth networks grow, and with them the ability to present quality video improves, the opportunity, need, and demand to produce video programs on a routine basis will also grow. It is already becoming necessary to reduce or eliminate the manual components of routine or ad hoc programs. Turning an auditorium into a Smart Space with the mission of capturing the talks that take place there is a natural way to supply those programs.

Our own experience shows that having an AutoAuditorium System allows us to record and broadcast programs that otherwise would not have been captured.

References

1. Bellcore Information Networking Research Laboratory (Mauricio Arango, Lisa Bahler, Peter Bates, Munir Cochinala, David Cohrs, Robert Fish, Gita Gopal, Nancy Griffith, Gary E. Herman, Takako Hickey, K. C. Lee, Will E. Leland, Carlyn Lowery, Victor Mak, John Patterson, Lillian Ruston, Mark Segal, R. C. Sekar, Mario P. Vecchi, Abel Weinrib, and Sze-Ying Wu), "The Touring Machine System", *Communications of the ACM*, Volume 36, Number 1, pages 68-77, January 1993.
2. D. V. Rabinkin, R. J. Renomeron, A. Dahl, J. French, J. Flanagan and M. Bianchi. "A DSP Implementation of Source Location Using Microphone Arrays", *J. Acous. Soc. Am.*, Vol. 99, No. 4 Pt. 2, p. 2503, April 1996 (131st Meeting of the Acoustical Society of America, May 1996).
3. D. Rabinkin, R. Renomeron, J. French and J. Flanagan. "Estimation of Wavefront Arrival Delay Using the Cross-Power Spectrum Phase Technique", *J. Acous. Soc. Am.*, Vol. 100, No.

The Personal Node (PN)

Gregory G. Finn and Joe Touch
USC/Information Sciences Institute
{finn, touch}@isi.edu
July 27, 1998

ISI/RR-98-461

Abstract

A Personal Node (PN) is a small, wallet-sized device that integrates people into the Internet. A PN incorporates wireless communication, limited user I/O, and local environmental telemetry to catalyze the coordination of other smart space and network devices for the user's benefit. By themselves smart spaces are not aware of the people in them and people are not aware of what is in a smart space. The PN allows the smart space to interact continuously with a person, and a person to interact continuously with the space, mediating the interaction with the help of other devices throughout the system. A PN is an individual's networking focal point. As the user roams about, a PN persistently maintains user presence on the internetwork. This represents the final and missing link in smart spaces, bringing the user in as a system resource and participant.

1: Introduction

In the near future buildings, rooms and vehicles are expected to evolve into *smart spaces* that contain varieties of wireless *smart devices*. As individuals and vehicles roam they will swim in a virtual sea of such devices. Those spaces must become aware of the individuals in them and vice versa. A Personal Network Node (PN, or Personal Node, for short) is a small device that is continuously carried by an individual, **allowing the user to become a permanent part of the smart space**, and allowing the infrastructure ongoing access to the user for feedback and commands. The PN is the network corollary of the PC; it is a network node for the 'rest of us'.

A PN incorporates promiscuous wireless interfaces to allow it to act as a networking catalyst. PNs also include various telemetry sensors, such as location (GPS), temperature, and orientation. This allows a PN to be addressed by-interface, by-region, by-proximity or by-heading. The ability to form ad-hoc networks coupled with multiple addressing modalities offers considerable benefits for military, commercial and emergency services.

1.1: Need for a Sixth Sense

Wireless technology allows us to create smart spaces and devices; key to its success will be its ease of use. People do not have the capability of directly interacting with smart spaces. We are deaf, dumb and blind as far as smart spaces are concerned, and won't be aware when we are in a smart space or know what's in one. Conversely, smart spaces won't be aware of our presence. This mutual lack of awareness must be overcome before smart spaces can become well integrated into our daily lives.

In the distant future, smart spaces may include senses so that they can be aware of and interact directly with us. A more practical solution is for us to become aware of smart spaces and interact with them on their terms, and to extend them to interact with us continuously on ours; in effect, to carry that sense with us. We need to acquire a sixth sense that allows us to continuously inhabit, to 'see and speak' in this wireless spectrum.

One way to make us aware of our wireless surroundings as we roam is to carry a device that intermediates with smart spaces on our behalf. PNs see and speak for us in those areas of the wireless spectrum used by smart spaces.

This document outlines our vision of the PN, and how it uniquely enables smart spaces interaction via a continuous network presence for people. The remainder of this document is organized as follows:

- *Sec. 2: Defining a PN*
- *Sec. 3: Need for a PN*
- *Sec. 4: Research Issues*
- *Sec. 6: Related work*
- *Sec. 7: Implications*
- *Sec. 8: Summary*

1.2: A vision of smart spaces

The following example describes our vision of the future of smart spaces, and how it is affected by having people as a continuous part of the infrastructure.

Although this scenario superficially has much in common with similar visions (e.g. Active Badges), a fundamental distinction underlies the PN: this device serves as the user's "eyes and ears" to the Internet (i.e. the sixth sense). It is not intended to be a compute node, but rather the minimal set of I/O functions that a human needs to interact digitally.

The guest arrives and walks through the hotel lobby, when his PN transfers its proxy to a local processor in the hotel. The transfer disconnects his previous proxy, in the car, and informs his base location, used as a rendezvous point. Messages for this guest are rerouted to the hotel.

The hotel provides the PN with a digital building map that includes emergency exit routes, local emergency services public keys and contact information. This data is particularly useful in emergencies such as fires and earthquakes, where, together with its own and the building's integrated sensors, the PN acts as a guide dog, leading the user to safety, or relaying his location to rescuers. Caching this data in the PN itself ensures fail-soft operation in the event of severe local service outages.

Simultaneously, the hotel smart space authenticates the guest's identity with the PN, his room air conditioner is activated, and door lock enabled with a new entry code. The PN adds the hotel's room-code index name and key value to its utility smart card, alerts its owner and tells him his room number.

In the vision just described, the PN alerted the smart space of the user's presence, and allowed another user (via her PN and its proxy) to alert him an urgent query. This interaction did not require external terminals; simple interactions can be handled directly by the PN.

2: Defining a PN

A PN is designed to be a networking *vade mecum*¹ that is active whenever you are. It continuously maintains your presence in the network, and contains sufficient I/O capability to allow you to communicate until other resources are arranged.

1. *vade mecum* (latin) - lit. "go with me"

2.1: Basic Assumptions

For PNs to function successfully a number of conditions must be met. Most amount to assumptions regarding the way the Internet will evolve. We state these here:

- *Internetwork wireless services will become ubiquitous. We expect most buildings and metropolitan areas to provide such service.*
- *In-building networks will provide a basic level of Mobile IP [24] service to users that enter their smart spaces. We also assume ISPs will support Mobile IP.*
- *Exterior and interior wireless networking technologies will be different and will provide different levels of service. For example, in-building wireless systems may offer greater bandwidth and lower latencies than that provided by exterior wireless systems.*
- *Smart-space computing will evolve to be increasingly 'user state' sensitive. We expect awareness of the users in a smart space and telemetry from them to become increasingly important.*

2.2: Design Principle and Content

The main design principle of the PN can be expressed as, "have only those capabilities that cannot be moved elsewhere":

- *Enough user I/O to bootstrap the user-network interaction.*
- *All the I/O that's particular to the user's locale.*
- *Support for the above, i.e., wireless links, processing and memory for local operations and to bootstrap coordination elsewhere, and a battery that requires recharging "when the user does".*

A PN is implemented as a hand-held sized device, including:

- *A variety of wireless interfaces:*
 - Fast IR, for Mbps desktop roaming
 - Wireless LAN for 1 Mbps office roaming
 - Cellular for 100 Kbps MAN roaming
 - Satellite wireless for WAN roaming
- *A limited amount of user I/O:*
 - microphone, speaker
 - small LCD (PDA or smaller)
 - buttons (3-4, re-labellable; or a touchscreen)
- *As much telemetry I/O as possible:*
 - orientation: GPS, accelerometers, compass

- environment: temperature, humidity, light / IR, (sound via mic), camera, EMI, pH, other chemical sensors
- personal biometry: pulse, respiration (via sound processing), blood pressure, fingerprint

- *Support for the above:*

- smart-card socket
- CPU and memory (volatile and non-volatile)
- battery for 48+ hours continuous "operation"

A PN is not a workstation or laptop. It does not include:

- *File storage:*

- Fixed disk
- Removable media

- *Traditional I/O:*

- Keyboard and display

A PN mediates on behalf of its user with smart spaces that it encounters, making these smart spaces aware of the user and the user's environment. Conversely, a PN becomes aware of what is in the smart space environments. A PN also caches information of immediate or local relevance on behalf of the user.

To achieve these objectives, PNs must be carried by their users most of the time, much as pagers and cellular telephones are today. To be comfortably carried a PN must be small and lightweight. This precludes its having a standard keyboard or display. Unlike a wearable computer or laptop, a PN is not intended to replace a workstation [30]. It is assumed that a user's primary computing and storage resources are located elsewhere, and the PN acts as a bootstrap to catalyze the coordination of these other resources.

A PN contains I/O peripherals that are intrinsic to an individual and that cannot effectively be located elsewhere. This includes user I/O, including microphone and speaker, control keys, and a limited display. It also includes sensors that monitor you and your local environment. Sensors would include GPS to determine your location, electronic compass, accelerometers, photometer, barometer and biometry. Unlike PDAs, PNs are IP-addressable, operate continuously, autonomously and communicate with their surrounding wireless environment.

There are also a variety of research issues involved in the development of a PN. Primary among these is power conservation, which drives hardware design as well as protocols to support multi-capability signalling. The continuous connectivity of the PN presents challenges to the traditional network model of a node (host), as well [4].

2.3: How is a PN different?

The general concept of a PN is related to that of the ParcTab. The PN is different from either a ParcTab or PDAs in that it:

- *Supports continuous internetwork presence*
- *Is I/O rich: multiple wireless interfaces, audio, biometry and environmental instrumentation*

The PN extends the capabilities of smart spaces, allowing them to include participants outside the conventional range of interaction. It is the continuous inclusion of the user as a node in the global infrastructure that uniquely distinguishes the PN.

3: Need for a PN

As has already been discussed, a PN helps integrate people with the network infrastructure, enabling new uses for smart spaces. This section elaborates on how PNs help complete the network component design space, and examines the variety of capabilities it enables.

3.1: PN as a "Smart Spaces Device"

Since the earliest days of the ARPANet, networking has been based on the notion of two kinds of components - hosts and routers [4] [6]. This model has been modestly extended via Mobile IP and DHCP to support hosts that relocate periodically, such as laptops. IP telephony also seeks to introduce a new kind of node into the Internet, i.e. the IP telephone.

New types of devices are extending this model even further. The notion of a host as an aggregation of host- and router-like devices is the basis of desktop area networks (DAN [1], Viewstation [14]). In this case, the previous model of host as a single, terminal point on a network is insufficient. The network of workstations has become another example of this kind of virtualization and aggregation.

Even so, the most novel recent network devices are expressible in this model. Wearable computers are typically modeled as laptops, where they are not worn continuously, and move from network to network only sporadically. They are designed more as a sophisticated PC, one that replaces the conventional desktop workstation, but is intended as the same kind of stable network device.

By analogy to the telephone system, the PN can be shown to fill a gap in the design space of network devices. There have been earlier attempts to address this 'non-host, non-router' role [27]. In the early days of telephony, devices were scarce and expensive, so party lines (e.g.,

mainframes with batch sharing) were the norm. As telephones became less scarce, direct dial and per-home installations became the norm (PCs).

Users began to rely on access to telephones, due to their pervasiveness in permanent installations. They extended this demand to mobile use, initially by "move and reconfigure", e.g., mobile telephones requiring operator assist and pre-configuration.

Up to this point, message services and answering machines were necessary, because the desired party wasn't always available when called. Telephone use was significantly asynchronous, except during business hours.

The telephone system eventually evolved to support true mobile phones and pagers. Both pagers and mobile phones are both ubiquitous and present with the desired person, with a single, persistent number. Prior to this, a phone number was a business or home, or car or boat. Now phone numbers are, in effect, "people".

Network hosts have evolved from 'office' addresses to laptop addresses. The next step in this progression is the PN, which allows people to become nodes.

3.2: PN Application Domains

There is a phase change in behavior when accessibility is continuous, as has been seen in the use of pages or cell-phones. The PN makes networking access continuous, makes smart spaces aware of the user and provides feedback on current user state. By design the PN is a catalyst enabling personal interaction with smart spaces.

We present a variety of examples of how a PN catalyzes personal interaction with smart spaces. These include:

- *Presence Sensitive Applications*
- *Smart Emergency Spaces and Services*
- *Autonomous Information Gathering*

Presence Sensitive Applications

The earlier hotel example demonstrated how sensitivity to user presence can be used by a smart space. There will be many such applications.

- *Security system smart doors could be automatically unlocked for authorized individuals.*
- *A smart docent notices a PN in its space, asks the PN its user's language preference and level of subject knowledge, then begins delivery of personally tailored audio program.*
- *Soldiers' PNs are interrogated for equipment and supply levels. Such information would be aggregated and fed back to logistics support.*

- *A smart transportation space would collect user destination information, e.g., to skip subway stops where no users are waiting or want to depart*
- *Upon entering a smart business space PNs could be supplied with the businesses web pointers. Current prices, not listed on the public web site, could be provided explicitly to local PNs.*
- *In training scenarios the status of individuals entering a smart space could be ascertained and their behaviors captured.*

Smart Emergency Spaces and Services

A PN is a wireless point-of-contact for the individual carrying it that contains instrument and computational assets. This combination enhances emergency services.

- *A PN can be loaded with site-specific emergency services information and software.*
- *A PN can verify the authenticity of emergency-services messages.*
- *When local communication services fail, a PN can draw upon its other wireless interfaces for support, creating an ad-hoc base-less network*
- *Because a PN knows its location, it can be addressed by location and proximity, locating an individual or limiting alarms to affected PNs*

Autonomous Information Gathering

As a user roams, their PN will pass through smart spaces. Information that may prove useful could be regionally broadcast to passing PNs. The reverse operation, in which the PN broadcasts information into a smart space, is also useful.

- *PNs located in vehicles or carried by soldiers could respond to queries regarding current level of supplies. This data could be aggregated and fed upwards to logistics personnel for disposition. If a PN indicates a shortage of critical supplies, the driver or soldier could be directed to the nearest depot location that stocked those supplies while also debiting depot inventory level.*
- *The ability to locate an employee and their vehicle allows delivery, repair and pickup services to operate more efficiently. This also allows resources to be staged as people move, extending the smart space to a smart logistics capability.*
- *In a commercial setting, PNs would anchor a broad new set of autonomous customer services. For example: A PN can automatically identify its user, allowing the business to use profiling information to direct the user to particular specials.*

4: Research Issues

PNs provide fundamental and new capabilities that have not previously existed. Their use raises a number of research issues concerning protocols, naming and addressing, coordination and configuration, privacy, scale and user interfaces. Their existence makes possible entirely new application areas that involve roaming, information capture and ad-hoc networking.

4.1: Protocols

The PN is predominantly a communications device, so the main research issues relate to protocols.

Integration with existing protocols

Many of the current Internet protocols are challenged by the way in which PNs change the model of a host. The PN requires continuous communication as it spans different link technologies, and may require periodic hibernation to conserve power. State-oriented network protocols, notably TCP, do not react well to such idle periods, and are not intended to support endpoint renaming. TCP also requires several round-trip times per exchange, increasing the latency of simple request/response protocols. Newer variants, including T/TCP, reduce this effect, but require further modification to support seamless transitioning [5].

The PN may also need to support proxy operation of other protocols. These might include delegated request/response, where the request is issued to a smart space resource which coalesces responses on behalf of the PN. Other remote protocols may include switchboarding, or remote control of a variable delegation point, such as to coordinate or redirect multimedia streams among other smart space devices.

Device-Control Protocols

Smart devices and PNs are network attached peripherals (NAPs). NAPs are relatively uncommon today and most utilize media-specific transport protocols. To avoid recreating the tower of babel problem that now exists with NAPs, smart devices should support Internet rather than media-specific transport protocols [13]. This requires the creation, testing and standardizing of physical device-control protocols and their APIs.

Because peripheral devices themselves fall into classes of similarity, such as disks and displays, it is reasonable to propose standardizing large portions of the lowest-layer interface that a class of smart device presents.

Internet accessibility increases risks to device privacy and integrity. Both privacy and integrity concerns should be met by adopting a methodology similar to that of Derived Virtual Devices (DVDs) [28].

Link agility

Roaming will move a PN into and out of contact with wireless networks. To maintain network presence the PN must monitor the 'liveness' of its links and associate with new ones as needed. Liveness monitoring and link association algorithms need to be developed and investigated. Low-power consumption is a requirement that must guide this work.

There are a number of triggering events to consider:

- *Failure of 802.11 low-power beaconing*
- *Hearing mobility agent from another subnet*
- *Distance from wireless hub and heading*

Tracking the rate of damaged messages combined with characterization of expected error rate versus distance [22] could also trigger association with a new network. Geographic information and application requirements may be of use when there is a choice of foreign agents and networks with which to associate.

Proxy protocols

The PN is not necessarily the best device to run request/response or stream protocols. Its limited bandwidth and power hinder it from first-class participation as a router in the smart space. Instead, it may be appropriate to off-load some protocols to proxies at other smart space devices.

These proxies could scatter requests and gather responses, effecting nested transactions. They could coordinate ordering protocols, or redirect continuous stream multimedia traffic, switchboard-style.

Protocol trade-offs

There are a number of bandwidth vs. latency vs. power trade-offs in a PN, some already presented. In general, wireless sending costs 10x the power of receiving, so the PN is a good platform for asymmetric protocols. Periodic retransmission of popular data, or server anticipation of PN requests based on traces, both can greatly reduce the power requirements and latency of access.

Furthermore, the bandwidth in and out of the PN is highly variable, and highly asymmetric as well. It may be feasible to enable the receiver of high-speed IR, but only the transmitter of the medium-speed office network. These trade-offs also affect the use of CPU and memory resources, because both can increase latency and power utilization.

4.2: Naming and addressing

The PN also challenges the traditional network notions of naming and addressing. The name of a PN should be intrinsically linked to that of the owner, not particularly to the device itself. GSM cell phones have this property,

where the GSM encoding card identifies the telephone number, independent of phone. However, there are other challenges, some of which are only beginning to be addressed in the research community.

Geographic Addressing and Broadcasting

Providing each PN with knowledge of its location allows it to be addressed via two modalities, by-interface and by-location. Geographic broadcasting is a particularly attractive new capability. Various approaches to realizing geographic broadcasting need to be examined as do protections against its unauthorized use.

Geographic routing and addressing in the Internet has been approached by creating a virtual network from geographically aware routers located within the Internet in a manner analogous to the initial implementation of multicast routing [16][21]. Earlier work pointed out that geographic addresses could be used to route packets in a hierarchic network and could support host mobility [11].

Geographic addressing could be grafted into IP as an option. However, IPv6 reserves 1/8th of its 128-bit address space for geographic addressing [12]. The possibility of providing hosts both interface-oriented and location-oriented addresses needs to be investigated, as do questions of the affect of geographic addressing on routing and transport protocols.

Smart Space Discovery

As a PN enters a smart space it should become aware of that smart space and vice versa. Mechanisms are also needed to discover what smart devices are in a smart space, which ones the PN can access and which application-layer interfaces they support.

Generalizing Multicasting

A geographic address naturally lends itself to description of a region. For example, a set of geographic addresses defines the interior of a polygon. A radius around a hub's location also defines a region. Such regions can be used to create routing tables [11] or to define multicasting groups [16][21].

Under the current Internet multicast routing scheme, a portion of the IPv4 and IPv6 address spaces are reserved for multicast addresses. Multicast groups are explicitly created and associated with a multicast address. A host explicitly joins a group to become its member and routers alter their routing tables to service group members. Once joined, a host remains a member until it explicitly leaves the group or the group is destroyed. Host mobility has no effect on membership in this type of group.

A multicast group could also be defined geographically. Examples are those hosts within a geographic region, a

room, a building, and so on. Ignoring for now precision of location, a host is either inside a group's region or outside it. Under this definition, membership in a geographical multicast group is implicitly determined. Host mobility does affect membership in this type of group.

The are great differences in how these two classes of multicast group are defined and in how membership is defined. Mobility makes ephemeral the membership criteria in geographically defined multicast groups. The set of such multicast groups that a mobile host is potentially a part of could change frequently. Existing Internet multicasting is ill suited to geographically defined multicast groups.

4.3: Coordination and configuration

The determination of resources in a smart space that a PN has entered and the run-time matching of those to application requirements is a producer/consumer problem. Solving this producer/consumer problem effectively is a major research task.

The question of what the appropriate OS will be for a PN also arises.

Configuring Smart Devices

In conventional system architectures, both device controllers and devices are resident in the chassis. The addition or removal of a system device is a relatively rare occurrence that can occasion rebooting of the system.

By way of contrast, in a smart-space environment the set of devices that a PN may come into contact with as the user roams will be large and dynamic. It will be unrealistic to expect a PN to be preconfigured with drivers specific to the smart devices that it encounters. This immediately imposes requirements upon PNs.

- *Need for Dynamic Reconfigurability*
- *Interface Matching*
- *Push/Pull Configuration Software*

4.4: Scale

Both PNs and smart spaces increase the scale of networks in a variety of ways.

The dynamic range of bandwidth increases, mostly due to a lower bandwidth required for WAN signals to the PN.

The latency range increases, also because WAN signals are liable to use satellite paths. These paths are already used in networks, but PNs would make them prevalent.

The number of devices in the network increases because addresses are required for smart space components, and they are required for PNs. The smart spaces

might support address aggregation, but the global roaming capability of PNs may inhibit such a simplification.

4.5: Security

There are two levels of security required for PNs. The data content itself must be secure (authenticated or encrypted), and the event of communication may also require privacy (source confidentiality). The latter would otherwise permit tracking, and behavior patterns themselves may constitute a compromise.

The need for security is especially important for the PN because it is so directly associated with a single individual, and because it contains so much local state (GPS, microphone, etc.).

There is an external aspect of security, that of device theft, which is also important. The small size required for vade mecum operation also encourages theft. Cell phone designers have adopted an automatic lock, which must be re-keyed with the user's PIN every time the phone is powered on. The PN might additionally require the PIN be re-entered on a schedule, every day or so. Much of the identity of the user, and much of her state, may also be encoded on a secure card, as in PCS cell phones.

Other devices may be used to simplify or augment the security provided by a PIN. These include biometrics (voiceprint, signature, fingerprint, other personal telemetry). Single-chip fingerprint imager chips are now emerging as commercial products along with the software to perform recognition.² Logging in would then consist merely of picking a PN up, pushing a button and swiping a finger across a scan pad. If a PN is used by a pool of people, the minutiae from prints of multiple individuals could be stored for recognition.

4.6: User interface

Size limitations prevent a PN from having a traditional keyboard or display. It is also unrealistic to expect individuals to use a conventional display and keyboard while roaming. Consequently, the user-interface for a PN must be nearly hands-free, depending primarily upon voice control with limited use of buttons.

Developing an effective nearly hands-free user interface for roaming will be a major research area. The need for speech recognition does imply that a PN needs access to significant computing resources.

When not roaming the user could pick up the PN and use it as a 3-D mouse, insofar as a PN contains accelerometers. Use could also be made of its small display. How-

ever, it is our contention that when users are not roaming they will likely be able to use nearby conventional display/keyboards.

Designing a user interface for a small display was part of the ParcTab effort [29]. We expect the ideas developed there would prove useful, but we view the small PN display as ancillary.

5: Feasibility

The PN is currently feasible, even given our demanding combination of capability, portability, and continuous operation. It is possible to bootstrap its development with an off-the-shelf, rapid prototype, in parallel with the coordinated application of well-known low-power, integrated packaging design.

5.1: Rapid prototyping

The principal components needed to prototype a PN are readily available. Setting aside the size and packaging issues, much of the needed research and development of the system architecture, protocols, and user interface could be done using a prototype built from off-the-shelf components. Existing PDAs and handheld PCs, together with wireless PCMCIA network interfaces can be used to emulate a PN. The PN must be more conservative in capability, to provide continuous battery-powered operation, however.

Once that prototyping effort is finished, packaging, size and power consumption issues could be addressed separately. The industry has already developed suitable low-power processors, memories and interface circuits. Examples are the SA-1100/133MHz microprocessor from Intel/Digital [9] that consumes 550 mwatts and non-volatile, zero-power SRAM.³

5.2: Power conservation

PDAs achieve their month-long recharge intervals by remaining normally off. They await an explicit activation by the user, perform a small amount of resulting computation, display the result and then turn off again.

On the other hand, PNs must maintain persistent network presence and so cannot be turned off. PNs must achieve a minimum recharge interval of 24 hours.

Variations on paging and polling techniques let a PN approach the normally-off power consumption characteristic of a PDA without sacrificing its network presence at

2. Examples are the Thomson-CSF FingerChip, Harris Semiconductor Corp. FingerLoc and the Veridicom FPS100.

3. Dallas Semiconductor DS1250 Power Cap SRAMs incorporate their own 10-year lithium cells.

the cost of increased latency. Each technique requires the cooperation of wireless hubs.

- *Incorporate a low-power pager in a PN. Hubs page a PN when they have traffic*
- *PNs poll their hub. This has scaling drawbacks but is it simple.*
- *Have the hub prompt polling by specific PNs. This technique is adopted in 802.11 low-power mode.*

Allowing a PN when roaming to remain inactive most of the time, will extend battery lifetime to a reasonable recharge interval. Overall, our current estimate of a PN is 3 watts of continuous operation power, or 8 oz. of zinc-air batteries/day. Paging-based wake-up allows discontinuous operation and will reduce demand, requiring only a small, matchbox-sized lithium or NiMH rechargeable battery.⁴

5.3: Communication

The PN requires the integration of a variety of link technologies, from high-speed desktop to low-speed wide-area links. Current variants of these include FIR (fast InfraRed) for the desktop, 802.11 wireless ethernet for the office, Metricom for the city, and text paging for the wide-area. Most metropolitan and wide-area communication technologies include their own power support, capable of 24 hours of standby operation and 1 hour of continuous operation. In the case of pagers, batteries last weeks, and support continuous monitoring of a very sporadic data stream.

These different link technologies have widely varying bandwidth, latency, and reliability capabilities. The application protocols need to adjust to the available link capability, operating in loosely- or tightly-coupled modes as needed.

5.4: Packaging

Packaging issues include power conservation, thermal diffusion, ruggedness, and integration for miniaturization. Current sensor technology already supports component-level and chip-level versions of many of the devices proposed for the PN. Power consumption and heat buildup can be reduced by conventional techniques (power devices only when in use, or only periodically).

The overall package needs to be wallet-sized (or smaller), and 1 lb or lighter. There appears to be a com-

4. Alkaline - MnO₂ batteries have an energy density of 75 WH/lb. Lithium - Li/MnO₂ batteries have 105 WH/lb and Zinc-Air 140 WH/lb. Source: <http://www.duracellnpt.com>.

mon *vade mecum* size, that of a cell phone or PDA, that is acceptable.

6: Related work

The PN is a variant of PDA technology and wireless 'presence' devices. It also extends the networking efforts of recent wireless and mobile protocols.

6.1: Handheld PDAs

Other more recent PDAs and handheld PCs have integrated small displays, touchscreens, and sometimes keyboards to provide rapid user access to limited local resources. Examples include the 3Com Pilot and a variety of WindowsCE palmtop PCs. The Philips Nino⁵ incorporates a microphone to support speech-based commands, in addition handwriting or script recognition provided by each of these devices. Compared to the PN, these palmtop examples lack environment sensors, and have only limited wireless capability (usually only IR on-board). More importantly, they are designed to be used only intermittently, in an "on, check/enter data, off" cycle lasting minutes. Their battery life is often measured in hours of runtime; they achieve weeks of life by being 'off' most of the time.

6.2: Wireless 'presence' devices

The ParcTab [29] was an early example of a wireless personal node. It had a single infra-red interface and provided persistent presence for in-building roaming. The PN extends the range of the ParcTab, supporting wireless access beyond the building confines. The ParcTab provided PDA-like services directly, whereas the PN focuses on catalyzing of other devices on its behalf.

The Lovegety is a simple wireless personal node that demonstrates mobile information capture [17]. It uses peer-to-peer beaconing to allow singles to meet when in one another's proximity, by exchanging a few bits worth of preference information. The device is small enough to fit on a keychain, and runs for days to weeks on battery power. This simple device can be considered a low-bandwidth variant of a PN, enabling singles to detect each others' presence with a smart space created by a multi-party ad-hoc baseless network.

5. <http://www.nino.philips.com>

6.3: Wireless and mobile protocols

The problem of providing persistent internetwork presence while roaming is the subject of the Mobile IP development effort [24][18]. A mobile computing environment that uses multiple types of wireless networks is called a *wireless overlay*. Maintaining connectivity while running applications in this environment is extremely challenging [19][26] and is the subject of a number of research efforts, including BARWAN [2] and Odyssey [23]. Operating systems originally designed for workstations require extensive changes in a nomadic environment [3].

Imielinski and Navas proposed embedding of geographic routing and addressing in the Internet by creating a virtual network from geographically aware routers [16][21]. Geographic addressing can also be directly used to route packets, support host mobility and provide regional broadcast [11].

7: Implications

The concept of a PN extends and challenges smart spaces and general issues in network research. By including users as nodes in the network, it extends the scope of the network, and the capability of applications within it. It also provides an opportunity to apply technology being developed for low-power, integrated sensors in a unique way to provide ad-hoc mobile smart sensor nets.

7.1: Smart spaces

The PN avoids the distinction between a user's on-line and off-line presence. The user is always on-line, accessible to signal for feedback, supporting immediate urgent-mode interaction. Because of its integrated, multi-level communications links it provides an opportunity to catalyze the aggregation of other network resources for the user's benefit.

7.2: Network research

Traditional networking considers users as temporary presences at permanent end-points known as hosts. The PN extends this notion, where people themselves become end-points on the network. People are more mobile, even than laptops, and so require hand-off without dead time, and a truly persistent identifier. Location of a user becomes a key network resource discovery issue.

The PN provides an opportunity to review more conventional host and gateway requirements, using a model that challenges their assumptions. Overall network architecture, naming, addressing, and resource discovery all may require re-examination.

In addition, transport protocols may require additional support for continuous relabeling of the endpoints, as users move between smart spaces. The PN itself may provide bridging capabilities between adjacent PNs when necessary. Finally, the traditional request/response protocols may require redesign, to support a proxy-mode operation, to off-load capabilities to smart space resources and conserve local power.

7.3: Application of related technology

There are a number of related technologies that are required for a PN to be developed. Small, low-power sensors already exist, but need to be integrated with a small amount of processing and storage into a handheld device. The PN focuses on placing as much I/O technology where the user is as possible, so there is virtually no limit to the challenge to integration technology here.

By placing the sensors where the user is, the PN provides a unique opportunity for ad-hoc deployment of sensor networks, in effect a mobile smart space centered, and concentrated exactly where the users are. Deployment at the appropriate place is de-facto achieved.

There is also a challenge to integrate a number of wireless communication technologies into a single, low-power, configurable device. This includes bandwidths from 1-1Mbps, latencies from μ s to 100's of ms, and ranges from feet to tens of miles, using IR, CDMA, GSM, and even simple analog paging technologies.

8: Summary

Wireless technology will soon be used to create and leverage smart spaces comprised of peripheral devices and sensors that communicate with one another and the network. As humans we can't directly perceive the wireless spectrum and so we aren't aware when we are in a smart space and we won't know what's in it. Conversely, smart spaces won't be aware of our presence. As long as people do not have the capability of directly interacting with smart spaces as they roam, smart spaces will remain restricted in their scope of application and ease of use.

These issues are addressed by creating small personal wireless nodes (PNs) that are carried with individuals as they roam. The PN's goal is to integrate the human being into the Internet.

The PN allows the user to become a persistent part of the network, by providing:

- - *continuous communication with the user via minimal I/O*
- - *a variety of user-centric telemetry and biometry sensors*

By providing a minimal initial access, these capabilities can be used to bootstrap the user's access to more advanced services, and to support ad-hoc base-less networking when disconnected from the rest of the net.

Acknowledgments

The authors would like to thank Bill Manning, Gene Tsudik and Jon Postel for their helpful suggestions.

9: References

- [1] Barham, P., Hayter, M., McAuley, D., Pratt, I.
Devices on the Desk Area Network
IEEE Journal on Selected Areas in Communications, May 1995.
- [2] BARWAN Project
http://www.cs.Berkeley.edu/~randy/Daedalus/BARWAN/BARWAN_over.html
- [3] Bender, M., Davidson, A., Dong, C. et al.
Unix for Nomads: Making Unix Support Mobile Computing
First USENIX Symposium on Location Dependent Computing, 1993.
- [4] Braden, R (ed.)
RFC 1122: Requirements for Internet Hosts - Communication Layers
Network Working Group, Oct. 1989.
- [5] Braden, R.,
RFC 1644: T/TCP -- TCP Extensions for Transactions, Functional Specification
USC/Information Sciences Institute, Jul. 1994.
- [6] Braden, R., Postel, J.,
RFC 1009: Requirements for Internet Gateways
ISI, Jun. 1987.
- [7] Davis, C., Vixie, P., Goodwin, T., Dickinson, I.
RFC1876: A Means for Expressing location Information in the Domain Name System
University of Warwick, Jan. 1996.
- [8] Deering, S., Hinden, R.
RFC1883: Internet Protocol, Version 6 (IPv6) Specification
Xerox PARC, Ipsilon Networks, Dec. 1995.
- [9] DIGITAL Semiconductor SA-1100 Data Sheet EC-R8XUA-TE
Digital Equipment Corp., Maynard, MA., Jan. 1998.
- [10] Farrell, C., Schulze, M., Pleitner, S., Baldoni, D.
RFC1712: DNS Encoding of Geographical Location
Curtin University of Technology, Nov. 1994.
- [11] Finn, G. G.
Routing and Addressing Problems in Large Metropolitan-scale Internetworks
ISI Research Report ISI/RR-87-180.
USC/Information Sciences Institute, Mar. 1987.
- [12] Hinden, R., Deering, S., editors
RFC1884: IP Version 6 Addressing Architecture
Xerox PARC, Dec. 1995.
- [13] Hotz, S., Van Meter, R., Finn, G. G.
Internet Protocols for Network-Attached Peripherals
Proc. Sixth NASA Goddard Conference on Mass Storage Systems and Technologies and 15th IEEE Symposium on Mass Storage Systems, Mar.1998.
- [14] Houh, H., Adam, J., Ismert, M., Lindblad, C., Tennenhouse, D.
The VuNet Desk Area Network: Architecture, Implementation, and Experience
IEEE Journal on Selected Areas in Communications, May 1995.
- [15] Imielinski, T., Navas, J. C.
GPS-Based Addressing and Routing
Technical Report (LCSR-TR-262)
Rutgers University Computer Science, Mar. 1996 (updated Aug. 9, 1996).
- [16] Imielinski, T., Navas, J.
RFC2009: GPS-Based Addressing and Routing
Computer Science, Rutgers University, Mar. 1996.
- [17] Lovegety
<http://www.freeyellow.com/members2/onmarketing>
- [18] Johnson, D. B.
Mobility Support in IPv6
Carnegie Mellon University,
Sun Microsystems, (work in progress).
- [19] Katz, R., Brewer, E.
The Case for Wireless Overlay Networks
Ch. 23 in *Mobile Computing*,
Kluwer Academic Publishers, 1996.
- [20] Mockapetris, P.
RFC1035: Domain Names - Implementation and Specification
USC/InformationSciences Institute, Nov. 1987.
- [21] Navas, J. C., Imielinski, T.
GeoCast: Geographic Addressing and Routing
Proc. of the Third ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97), Budapest, Hungary, Sep. 26-30 1997.
- [22] Nguyen, G., Katz, R., Noble, B., Satyanarayanan, M.
A Trace-Based Approach for Modelling Wireless Channel Behavior
Proc. of the Winter Simulation Conference, 1996.

- [23] Odyssey Project
<http://www.cs.cmu.edu/afs/cs/project/coda/Web/docs-ody.html>
- [24] Perkins, C., Editor
RFC 2002: IP Mobility Support
IBM, Oct. 1996.
- [25] Rekhter, Y., Li, T., editors
RFC1887: An Architecture for IPv6 Unicast
Address Allocation
Cisco Systems, Dec. 1995.
- [26] Satyanarayanan, M.
Fundamental Challenges in Mobile Computing
Fifteenth ACM Symposium on Principles of
Distributed Computing, May 1996, Philadelphia, PA.
- [27] Stewart, B., (chair)
"Charter of the Special Host Requirements (shr)
working group",
Proc. of the 21st IETF, Jul. 1991.
- [28] Van Meter, R., Hotz, S. Finn, G. G.
Derived Virtual Devices: A Secure Distributed File
System Mechanism
*Proc. Fifth NASA Goddard Conference on Mass
Storage Systems and Technologies*, Sep. 1996.
- [29] Want, R., Schilit, B.N., Adams, N.I., Gold, R.,
Petersen, K., Goldberg, D., Ellis, J.R., Weiser, M.
The ParcTab Ubiquitous Computing Experiment
Ch. 2 in *Mobile Computing*
Kluwer Academic Publishers, 1996.
- [30] Wearable Computing FAQ, v1.0, 8/28/97.
[http://lcs.www.media.mit.edu/projects/wearables/
FAQ/FAQ.txt](http://lcs.www.media.mit.edu/projects/wearables/FAQ/FAQ.txt).

Multimodal Human/Machine Communication *

James Flanagan, Ivan Marsic, Attila Medl, Grigore Burdea, Joseph Wilder
Rutgers University
Center for Computer Aids for Industrial Productivity (CAIP)
New Brunswick, N.J.

Abstract - Natural communication with machines is a crucial factor in bringing the benefits of networked computers to mass markets. In particular, the sensory dimensions of sight, sound and touch are comfortable and convenient modalities for the human user. New technologies are now emerging in these domains that can support human/machine communication with features that emulate face-to-face interaction. A current challenge is how to integrate the, as yet, imperfect technologies to achieve synergies that transcend the benefit of a single modality. Because speech is a preferred means for human information exchange, conversational interaction with machines will play a central role in collaborative knowledge work mediated by networked computers. Utilizing speech in combination with simultaneous visual gesture and haptic signaling requires software agents able to fuse the error-susceptible sensory information into reliable interpretations that are responsive to (and anticipatory of) human user intentions. This report draws a perspective on research in human/machine communication technologies aimed to support computer conferencing and collaborative problem solving.

Networked computers are becoming pervasive. With this progress comes opportunity for accomplishing collaborative knowledge work by participants who may be geographically separated (Fig. 1). The networked system takes on the role of both mediator and aide, as it supports activities of increasing complexity. Through embedded intelligence and software agents, the system dynamically allocates computing and storage resources, as well as monitors quality of service. For example, network-congesting calls might be avoided if client objects and server objects are migrated to a common host prior to the calls (e.g., downloading of JAVA applets). Under this architecture and control strategy effective communication between the human user and the mediating system becomes central to realizing the full capabilities for collaborative effort.

* Based in part upon a plenary talk invited for the IEEE Workshop on Speech Recognition and Understanding, Santa Barbara, CA, December 14-17, 1997.

* Components of this research are supported through NSF Contracts MIP 93-14625 and IRI-96-18854 and through DARPA Contracts DABT63-93-C0037 and N66001-96C-8510.

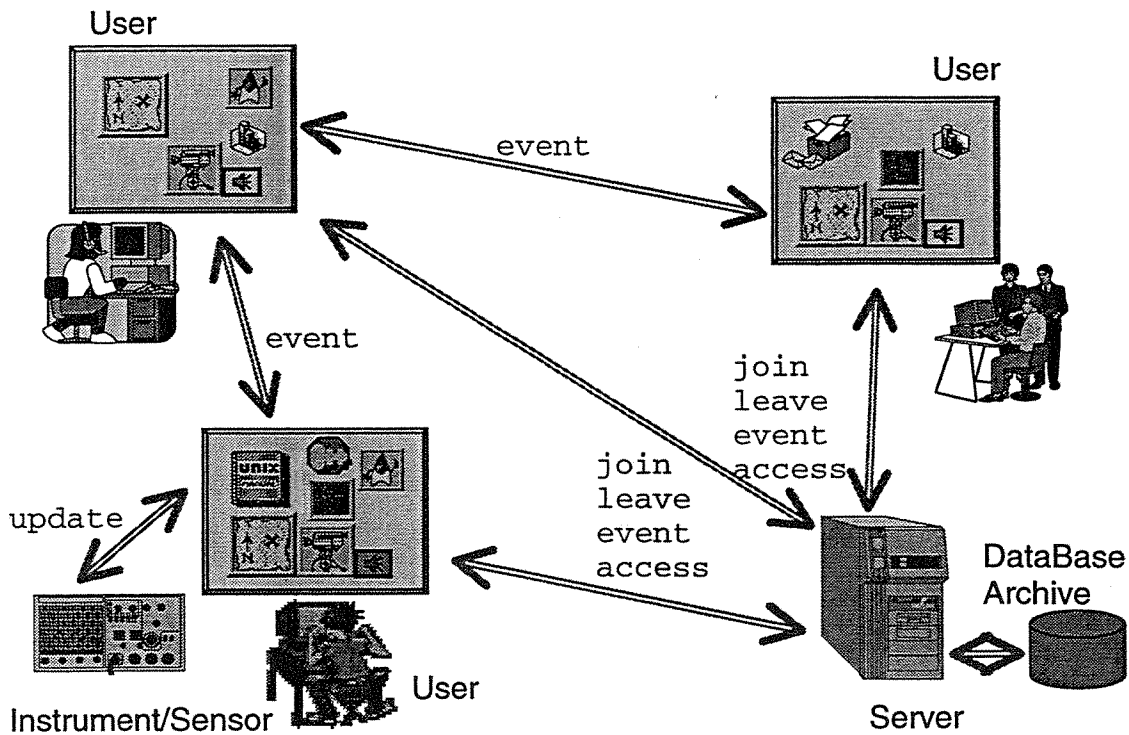


Figure 1 Networked computers with distributed processing, storage, archiving, and embedded intelligence support collaborative knowledge work. Probes installed in client and server Object Request Brokers monitor network traffic and automatically migrate objects whose frequent calls produce congestion.

Humans favor natural modes of communication for information exchange. The sensory dimensions of *sight*, *sound* and *touch*, used in combination, offer capabilities that go well beyond mouse and keyboard for collaboratively manipulating objects in a shared workspace. (Fig. 2). An obstacle is that the individual technologies for human/machine communication are, as yet, imperfectly developed. But prudently applied they can be used to benefit.

Prior research at the CAIP Center has established several interface technologies that, in properly delimited application, can contribute naturalness for human/machine interaction. They include:

- *sight*
 - region-of-interest segmentation
 - gaze tracking and visual gesture
 - face detection and recognition
- *sound*
 - beam-steered microphone arrays
 - speech and speaker recognition
 - computer voice response

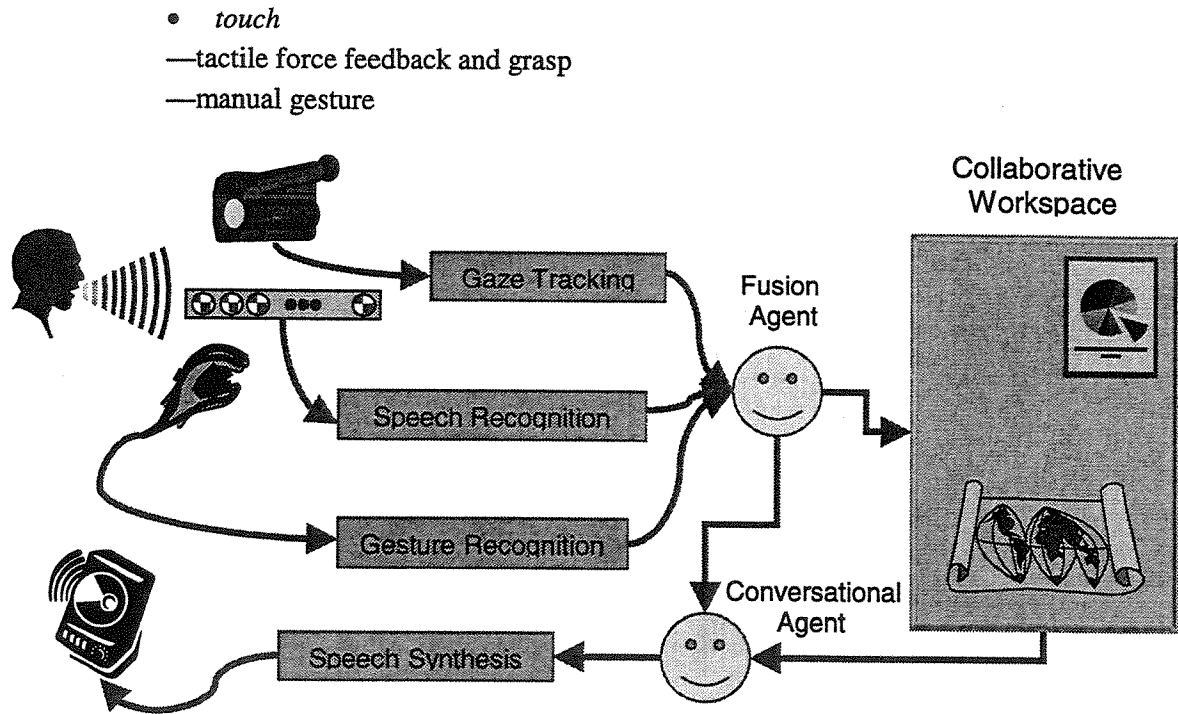


Figure 2 Interface modalities for sight, sound and touch — used in combination with intelligent data-fusion agents — provide enhanced, natural-like capabilities for cooperative manipulation of objects in shared workspaces.

Region-of-interest segmentation of images is included in an available suite of signal-processing algorithms which can be downloaded and locally executed. Automatic segmentation is based upon color, form and luminance, and is useful in identifying similar objects in complex images such as satellite pictures, blood-cell microscopy and MRI analysis. Gaze tracking provides an eye-controlled cursor derived from automatic face tracking and infra-red illumination (to determine the angle between the centroid of the pupil and the corneal reflection).

Speech recognition and text-to-speech synthesis permit conversational interaction with the system. Beam-steered microphones provide “hands-free” sound capture where the human is unencumbered by tethered or body-worn sound pickup equipment. Tactile force feedback is achieved from computer-controlled pneumatic thrusters applied to the finger tips of a close-fitting glove. Finger deflection is sensed by an LED and photo detector pair, located coaxially within each thruster. A Polhemus coil on the wrist provides absolute position.

Building upon these techniques, a current activity is the development of a client station that supports more natural communication with the user (Fig. 3). The ingredients include automatic face detection and gaze control of displayed cursors and objects, “hands-free” conversational interaction using speech recognition, synthesis and beam-forming microphone array, and manual

gesture and force-feedback grasp with a tactile glove. To employ these in combination (as the human does) the system must be able to fuse the often unreliable sensory inputs to reach a reliable decision and action that is responsive to the intent of the user (see Fig. 2).

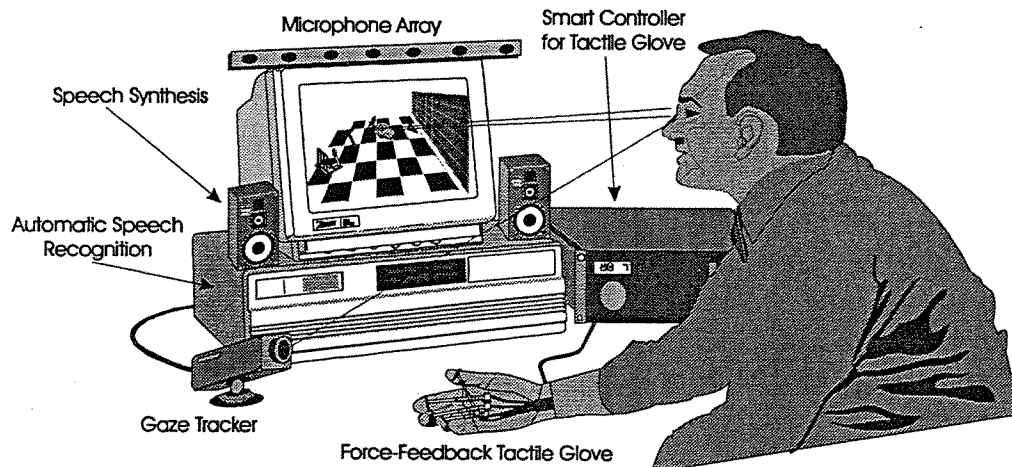


Figure 3 Experimental client workstation incorporating sight, sound and touch modalities for human/machine communication. The eye tracker provides a gaze-controlled cursor for indicating objects in the display. The tactile force-feedback glove allows displayed objects to be grasped, “felt”, and moved. Hands-free speech recognition and synthesis provides natural conversational interaction.

In an initial implementation we have combined gaze, speech, and tactile inputs to serve a specific application (i.e., civil preparedness and mission planning) that requires collaborative manipulation of graphical objects on a topological map displayed in the shared workspace. (Fig. 4). In this constrained task the vocabulary and grammar for the conversational interaction are delimited for increased reliability. The microphone array is integrated into the workstation housing and fix-focused on the user position (seated at the keyboard).

For the constrained task, fusion of data from the gaze, speech, and tactile inputs is accomplished by a slot filler method in which a parse of the recognized text string is synchronized with the visual and tactile inputs. (Fig. 5). The figure illustrates the command “move red circle to this location.” The recognized text string indicates the operation to be performed (“move”) and the object (“red circle”), but the destination (“this location”) is determined from either the manual gesture or visual cursor coordinates that temporally coincide with the utterance “this”. A preceding spoken command assigns the relevant cursor. Clearly, the sensory inputs can overlap in information content and exhibit redundancy. Clearly, too, in some instances a single modality is sufficient and natural, and can subsume the entire task. The fusion agent must cope with all combinations (a very natural utterance being “move this to there”).



Figure 5 Implemented user station incorporating speech recognition and synthesis for conversational interaction, combined and synchronized with manual gesture sensing from a force-feedback glove and visual gesture sensing from a desk-mounted gaze tracker. A fixed-focus microphone array atop the workstation captures speech from the user keyboard location while mitigating interfering acoustic noise and reverberation.

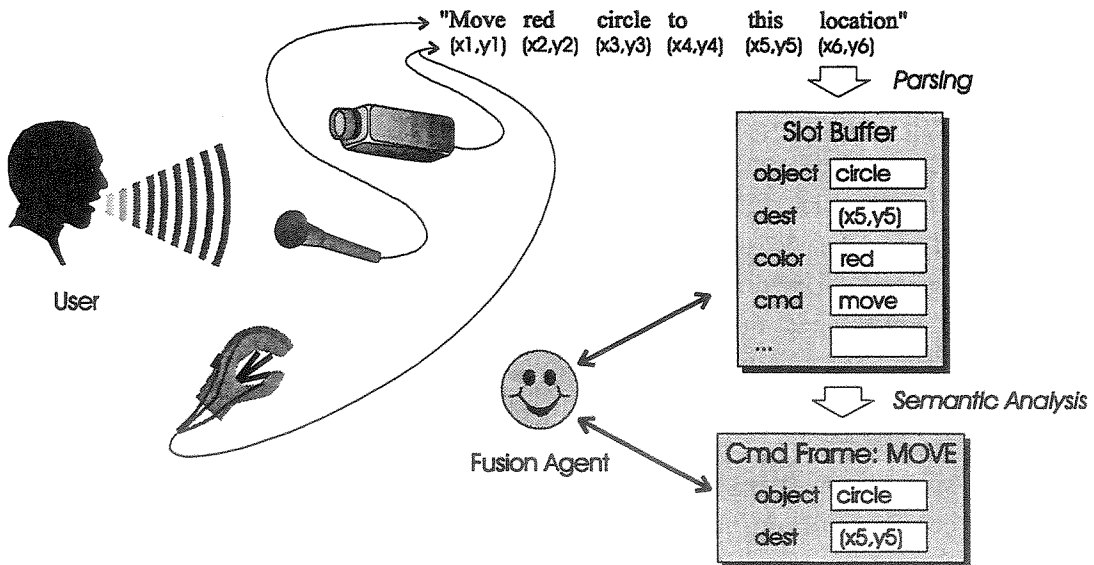


Figure 6 "Slot-filling" technique used to coordinate, fuse, and interpret simultaneous speech, visual, and tactile inputs.

For conversational interaction, synthetic voice answerback is essential so that the system can advise the user of actions needed or taken. Text-to-speech synthesis is the appropriate technology to supply the user:

- answers to queries related to the dynamic state of the workspace
- requests for confirmation if necessary
- general error messages and warnings about unexecutable commands
- notifications about semantic or user-related errors
- notifications about ambiguities and inconsistencies with suggestions on how they may be resolved.

The preceding has addressed single user communication at the client station. In some cases (such as situation rooms, command centers, or "smart-room" group teleconferences) multiple users must be accommodated at a client station. Speech sound pickup and face detection that can follow conferees about the meeting room are desirable under these conditions. Toward this capacity we have implemented an acoustic source locator which steers a line array microphone and points and focuses a video camera to the x,y,z coordinates of the source. (Fig. 6).

Source location is determined by two "quads" of omni-directional microphones placed preferably on adjacent orthogonal walls. The spacing is typically about 30 cm to insure that usefully correlated signals are obtained. For each pair in the quad an estimate of the time difference of signal arrivals is made. This estimate is obtained by computing an FFT for each microphone on a single DSP32C processor and forming the normalized cross power spectrum for each possible pair in the quad (i.e., 6). Inverse transformation gives the cross correlation function. A time difference of arrival estimate is made from the peak value. For the two quads, 12 such time difference of arrival estimates are made, each one defining the surface of a paraboloid on which a source could be positioned to produce the observed time difference. Ideally the intersection of the 12 surfaces would be a point. Practically, in most realistic environments it is a volume about 20 cm. in linear extent.

The 12 time difference estimates are passed from the DSP32C signal processor to a Pentium PC, which applies a gradient descent algorithm to find the x,y,z coordinates in the enclosure which provide the best least-squares fit to the set of time differences. (Computational speeds of DSP32C and Pentium presently limit the position determinations to 2 per second. Faster signal processors, under construction, aim to provide moving source estimates at 5 per second.)

Having made the x,y,z source location, the Pentium points and focuses a video camera to the source location. It also activates control hardware for an electronically-steered beam-forming microphone array of 21 first-order gradient microphones, pointing the beam to the source location.

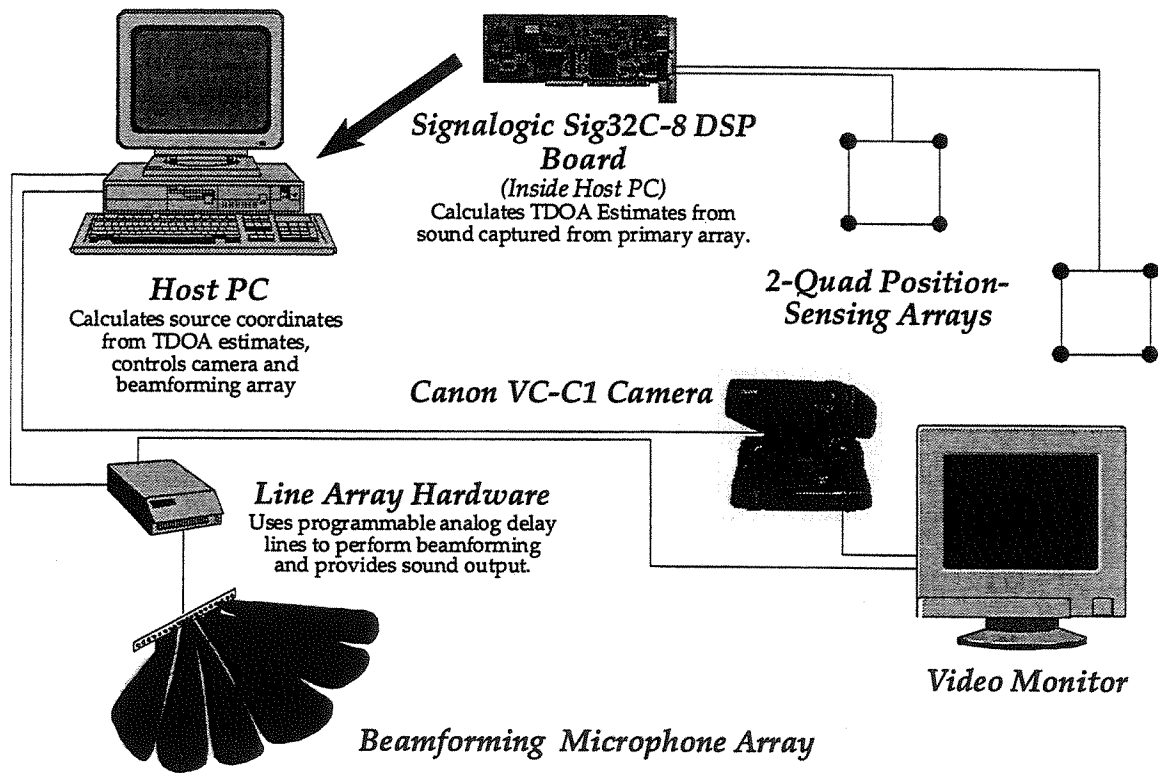


Figure 7 Auto directive system for image and sound capture in a “smartroom” client station. The system is able to locate and track the position of talkers moving in the conference room.

Finally, because tethered equipment is confining, a wireless appliqué is being considered for tactile gloves worn by multiple participants. Also for multiple users, personal identification is usually important. The data-fusing agent serving the client station may therefore need identity information on enrolled conferences passed to it from the interface modalities. The technologies of speaker identification and face identification are advanced enough to serve this need and can be used in combination to achieve high accuracies of identification.

References

1. J. Flanagan and I. Marsic, "Issues in measuring the benefits of multimodal interfaces," *Proc. IEEE Int'l. Conf. Acoustics, Speech and Signal Processing (ICASSP'97)*, Munich, Germany, April 1997.
2. A. Medl, I. Marsic, V. Popescu, A. Shaikh, M. Andre, C. Kulikowski and J. Flanagan, "Multimodal User Interface for Mission Planning," submitted to the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI'98), Los Angeles, CA, April 18-23, 1998.
3. G. Burdea, *Force and Touch Feedback for Virtual Reality*, John Wiley & Sons, New York, 1996.
4. J. Flanagan, "Technologies for multimedia information systems," *Proc. IEEE*, v.84, no. 4, pp.590-603, April 1994.
5. J. Flanagan and E. Jan, "Sound capture with three-dimensional selectivity," *Acustica*, v. 83, no. 4, pp.644-652, July/August 1997.
6. D. Rabinkin, D. Macomber, R. Renomeron and J.L. Flanagan, "Optimal Truncation Time for Matched Filter Array Processing", In *Proc. Of ICASSP'98*, v. VI, pp. 3629-32, Seattle, WA, May 1998.
7. R. Renomeron, D. Rabinkin, J. French, and J.L. Flanagan, "Small-Scale Matched Filter Array Processing for Spatially Selective Sound Capture", *J. Acoust. Soc. Am.*, v. 102, No. 5Pt. 2, p. 3208, Nov. 1997.
8. A. Waibel, M. Vo, P. Duchnowski, and S. Manke, "Multimodal interfaces," *Art. Intell. Rev.*, v. 10, nos. 3-4, 1995.
9. J. Wilder, P.J. Phillips, C. Jiand, and S. Wiener, "Comparison of Visible and Infra-Red Imagery for Face Recognition", In *Proc. Of the 2nd Int. Conf. On Automatic Face and Gesture Recognition*, pp.182-187, Killington, VT, Oct. 1996.

A NEW GENERIC INDEXING TECHNOLOGY

Michael Freeston

Alexandria Digital Library Project
Department of Computer Science
University of California, Santa Barbara
Santa Barbara CA 93106

ABSTRACT

We offer an enabling technology for information visualization and smart spaces in the form of a new indexing technology. We point out the limitations in both performance and functionality of the indexing methods found in the current generation of database systems, and sketch the properties of a radical alternative approach. Finally we give some background on the experience and qualifications of the author.

1. INTRODUCTION

We propose to contribute an enabling technology for information visualization and smart spaces in the form of a new indexing technology. The indexing method on which the whole of the database industry currently relies – the B-tree – has not changed for 25 years. We contend that this is inadequate and inappropriate for the kinds of application which it is now required to support, and that it is time for a radically new approach.

In its place we propose a new generic indexing technology, founded on a number of recent theoretical advances [Fre87, Fre89a, Fre89b, Fre95, Fre97], including a general solution of the n -dimensional B-tree problem – a problem which McCreight (co-inventor of the B-tree) described as “...one of the persistent puzzles of computer science” [McC85]. The theoretical basis of the technology is therefore now well established, and we have recently begun an NSF project to test the principles in practice.

The technology has two particularly relevant features: it is fundamentally multi-dimensional, and it has unique predictability and worst-case characteristics. So it is ideally suitable for supporting the organization and visualization of multi-dimensional data, whether explicitly spatial or not; and it is able to guarantee search times in real-time and mission-critical applications. It is also fully dynamic i.e. its performance does not degrade with updates, which is an essential property for non-stop systems.

Aside from the present focus of application, we believe this technology has the potential to dramatically improve the efficiency of conventional database indexing and querying. Nationally, even a 1% improvement would be worth billions of dollars a year in saved time. GIS and data mining applications stand to benefit particularly from better spatial and generalized multi-dimensional support.

2. LIMITATIONS OF CURRENT TECHNOLOGY

The smart spaces concept implicitly relies on a technological infrastructure which ultimately depends on very sophisticated information management systems. We submit, however, that none of the database or knowledge base systems existing today can convincingly support such a concept. To do so, a radical new approach to database indexing is required.

It is obvious that very fast update, access and display of *dynamic*, geo-referenced information will be necessary on a large scale. Every object - animate or inanimate - on the battlefield of the future will be constantly reporting its position and movements, and the position and movements of objects it observes. However, current technology supporting geo-spatial applications is still primarily geared towards Geographic Information Systems (GIS), which are essentially restricted to static geographic data. This fact may be hidden in impressive demonstrations of digital elevation map fly-overs, where the problem is to retrieve and display the data fast enough - but not to update it. Mountains rarely move.

A dynamic environment demands very fast, scalable - and of course dynamic - indexing methods. However, index updates of the currently most popular spatial indexing method - the R-tree and its R*-tree variant - are notoriously slow, do not scale and degrade with time.

Compounding these deficiencies, spatial indexing methods in existing database systems do not give the controlled worst-case access and update guarantees needed for mission-critical applications. Future self-guided vehicles, for example, must be able to predict the movements of other mobile objects around them in real time. This kind of application clearly differs from conventional GIS-type applications in that it requires the representation of a temporal sequence of events, and not simply a spatial distribution - however frequently such a distribution might need to be updated. Surprisingly little research has been devoted to temporal indexing of this kind within mainstream database systems, despite the fact that, even now, large-scale air-traffic control systems struggle with this problem.

Air-traffic controllers also struggle with information overload - a problem which must be addressed in a much more generalized way in a smart spaces environment. Yet commercial database vendors have so far been content to leave the filtering of data to higher level applications. This is highly inefficient. What is needed is *smart indexing* i.e. an approach to indexing in which

the selective retrieval of *relevant* information is built in to the index mechanism itself.

In particular, we see considerable potential value in *scalable queries*. Such queries return a set of results whose members conform to the same scale as the query. Thus a search for population centers within a geographical range enclosing the whole of the contiguous USA might return only those cities with populations over 1 million, whereas a query over a relatively small region might show towns down to 1000 inhabitants. (Note that 'scale' does not necessarily mean the same as 'size'. Politically, Texas and the District of Columbia might have the same scale). This is just one example of the way in which we think smart indexing can help to present "the big picture", free from irrelevant clutter or information overloading.

The very term *smart spaces* implies some kind of reasoning ability. This requires support for inferencing systems and rule bases - possibly very large rule bases. Their use may extend from strategic decision-making to the application of transitive closure algorithms to find optimum paths through logical or physical networks. A fundamental problem has always been - and remains - how to build a scalable, dynamic rule base, and how to integrate it with the data manager [NS88, SSU91]. Without a solution to this problem, we cannot hope to develop decision support systems capable of themselves suggesting or validating a course of action, rather than merely presenting the information on which decisions must be made.

3. FOUNDATIONS OF A NEW APPROACH

We have conducted an intensive research program addressing these basic indexing problems for the past ten years. The result is a new multi-dimensional indexing technology which revolutionizes the very principles on which conventional databases are founded, and offers a solution to the problem of how to build scalable and dynamic rule bases. It can index spatial databases up to two orders of magnitude faster than conventional spatial indexing methods.

The philosophy behind the new approach is in direct contrast to the current trend towards 'data blades' or 'data cartridges' i.e. the idea of plugging in a different index method for each data type. By studying the underlying principles of dynamic indexing techniques, we have been able to develop a *generic* method of indexing any data structures which can be expressed as directed graphs. This offers the potential for enormous software simplification and reuse in future data and knowledge base systems.

The new technology is based on a hierarchical index structure, like a B-tree [BM72]. But the similarity ends there. The most obvious difference is that the new tree does not appear to be balanced. What makes it absolutely unique among index structures is that it re-constructs itself dynamically during query and update, so that it *behaves* like a balanced tree. This is the essential trick which enables it to extend the worst-case search and update characteristics of the B-tree to n dimensions.

The fundamental advance is that this technique allows a true recursive partitioning of an n -dimensional data space. This makes it possible to propagate common prefixes of index keys all the

way to the root of the tree. The result is an extremely compact index with very short keys.

The keys themselves are simple binary strings. This is possible because of another fundamental distinction between the new approach and conventional database indexing: the index partitions the data space itself, and not the data values in the space. An important consequence of this is that 'empty' regions of the data space are not represented in the index at all. This 'pruning' of the data space means that searches which fail, do so very fast - the secret of constraint programming.

The index keys, which are of variable length, are generated by regular binary partitioning, cycling through the dimensions of the data space like a quad-tree - except that the resulting subspaces are not limited to squares. Another property that distinguishes the index from a quad-tree is that one subspace can enclose another. This is a consequence of the strange property that, although there are as many index entries as subspaces of the data space, an individual subspace may be represented by a set of index keys. This property - unique among indexing methods - turns out to be of fundamental topological significance, and is one of the main reasons why the new technique works.

As a result, it subsumes quad-trees, and for the first time makes possible a neighborhood-preserving mapping between a large-scale quad-tree and a balanced persistent tree structure. (It also offers novel ways of representing and storing images at different resolution levels). But the most important consequence of these properties is that the index keys do not, as in conventional indexes, *identify* individual indexed objects: each key contains just enough bits to *differentiate* one object from another in the index. This allows the compact representation of widely differing values in high-dimensional, sparse data spaces. Image feature classification is an ideal application area for this.

Finally, there is no reason why the index keys have to be restricted to the values of objects: they can also include bit-patterns to differentiate between objects of different structure. It is this property which allows the extension of the technique to the indexing of complex object structures, and deduction rules.

4. ABOUT THE AUTHOR

The author has long experience of both academic and industrial research, including almost 30 years of accumulated expertise in database research and development. During an initial period of teaching and research at Southampton University (UK), he built two relational database systems from scratch, and ran them successfully in very demanding commercial environments.

In 1985 he was invited to join the Knowledge Bases Group at the European Computer-Industry Research Centre (ECRC) in Munich, Germany. ECRC was set up in 1984 by a consortium of Europe's leading computer manufacturers - Siemens, Bull and ICL - as a European response to the challenge of the Japanese 5th generation project, and its US competitor at MCC.

The focus of research at ECRC was originally defined as the development of decision support systems based on logic, and the

mission of the Knowledge Bases group was to develop a large-scale deductive database technology. The author was involved in the design and implementation of a series of prototype systems which developed into one of the world's leading deductive database system research projects, and more recently evolved into the ECLiPSe persistent logic programming environment, now in use at over 300 sites world-wide.

A central problem was how to build large-scale, persistent logic programs, and it was this which led to the work on indexing methods for which the author is now internationally best known. The search for a solution to the original indexing problem began with multi-dimensional BANG indexing in 1987 and culminated in a general solution of the n-dimensional B-tree problem in 1995. There were many other advances along the way, but only now is it becoming possible to fit all the pieces of the puzzle together.

Because the problem turned out to be the most difficult of a wide class of indexing problems, a lot of valuable spin-off was generated which has spawned research in visualization, data mining, GIS and environmental management systems. The author has a long-standing interest in taxonomic database systems, contributing his database design expertise to the development of a computerized taxonomy (ILDIS) of all the world's legumes. ILDIS was the central inspiration behind the launch in 1996 of *Species 2000*, a major international initiative sponsored by the United Nations to enumerate all the plants, animals and microbes on earth by the year 2000. A further research direction, which arose originally from his contact with the research into constraint logic programming at ECRC, is the new field of constraint databases, to which he has been actively committed for the past three years as a member of the CONTESSA Working Group in the European Esprit program.

In 1990 he became Head of the Knowledge Bases Group at ECRC, responsible for the technical and administrative direction of fifteen researchers. In 1992 he became Head of Technology Transfer and Integration, responsible for the integration of all technology developed at ECRC and its transfer to ECRC's three parent companies.

In 1996 he moved to the US to join the Alexandria Digital Library (ADL) project at UCSB. This is one of the six digital library projects funded jointly by NASA, DARPA and NSF. The objective of ADL is to build a globally available digital library of geo-spatially referenced information. The author's contribution to its design is threefold: the development of a web-based distributed object model of the library; the provision and evaluation of new indexing methods; and the incorporation of large-scale knowledge base technology for decision support and data mining.

He is also currently involved in the planning of new proposals under the recently announced federal research initiatives on digital libraries (DLI2) and Knowledge and Distributed Intelligence (KDI). The central theme of these proposals is the move from passive to active use of the Web, the design of user workspaces, and the provision of user and system services - concentrating on digital library services and their integration into the wider concept of Distributed Knowledge Work Environments. His personal contributions to these proposals address the challenges of digital libraries as scaleable distributed object

repositories (i.e. repositories of data and software); efficient query support through advanced indexing methods - including "smart" indexing; deductive database technology as a generic platform for library knowledge bases and expert systems; and a database-supported open hypermedia system to create "trails" through collections of library documents.

In parallel with this research he has recently been awarded his own NSF grant to support further development and evaluation of his indexing technology, with digital libraries as the application focus. Results so far are extremely encouraging: the technology offers substantially better performance with lower software complexity over a wider range of functionality than existing techniques. It is currently being applied to multi-dimensional indexing in conventional databases, spatial and temporal indexing, data mining and visualization, text retrieval, and image feature classification. Future plans aim to apply the technology in the areas which originally motivated this research: deduction rules and complex objects in a persistent programming environment.

5. REFERENCES

- [BM72] R. Bayer and E. McCreight. *Organisation and maintenance of large ordered indexes*. *Acta Informatica* Vol. 1, No. 3, 1972.
- [Fre97] Freeston, M. *On the Complexity of BV-tree Updates*. ESPRIT-NSF Workshop on Constraint Databases and their Applications, Delphi, Greece, January 1997. [Lecture Notes in Computer Science No. 1191, Springer Verlag, 1997].
- [Fre95] Freeston, M. *A General Solution of the n-dimensional B-tree Problem*. Proc. ACM SIGMOD Conf., San Jose, California, 1995.
- [Fre89a] Freeston, M. *Advances in the Design of the BANG File*. 3rd Int. Conf. on Foundations of Data Organization and Algorithms (FODO), Paris, June 1989. [Lecture Notes in Computer Science No. 367, Springer-Verlag].
- [Fre89b] Freeston, M. *A Well-Behaved File Structure for the Storage of Spatial Objects*. 1st Symposium on the Design and Implementation of Large Spatial Databases, Santa Barbara, California, 1989. [Lecture Notes in Computer Science No. 409, Springer-Verlag].
- [Fre87] Freeston, M. *The BANG file: a New Kind of Grid File*. Proc. ACM SIGMOD Conf., San Francisco, 1987.
- [McC85] E. McCreight. *Priority Search Trees*. *SIAM Journal of Computing*, Vol. 14, No. 2, May 1985.
- [NS88] E. Neuhold, M. Stonebraker (Ed.). *Future Directions in DBMS Research*. Report no. TR-88-001 International Computer Science Institute, Berkeley, California, May 1988.
- [SSU91] A. Silberschatz, M. Stonebraker and J. Ullman (Ed.). *Database Systems: Achievements and Opportunities*. Communications of the ACM, Vol. 34, No. 10, October 1991.

Configuration Challenges for Smart Spaces

John Heidemann, Ramesh Govindan, Deborah Estrin

University of Southern California/Information Sciences Institute

ABSTRACT

Smart spaces will be composed of thousands of computing elements interacting with the physical world. We argue that automatic configuration and resource discovery remains an important unsolved problem in smart-space deployment.

1. INTRODUCTION

Trends in CPU performance and packaging now allow very powerful devices to be embedded on single chips. Today a car may have a half-dozen processors on-board that are monitoring brakes, improving fuel efficiency, and even regulating climate. We are poised to live in a world of *smart spaces* where most things around us (from bandages to books) have computing power and interact with the world.

Until recently smart spaces have been limited by computing power, price, and energy. Communication between nodes has been extremely limited. Moore's law has attacked computing power and price, a variety of initiatives are attacking energy. Finally, the growth of wireless networking hardware and explosion of the Internet promises COTS networking hardware and protocols. Or do they?

Although many challenging impediments to smart spaces have been overcome, we believe direct application of COTS networking protocols will not prove satisfactory. Today's networking infrastructure has many features important in connecting smart spaces: Internet protocols and implementations are open, widely implemented, support multiple physical layers, and are relatively efficient and lightweight. All of these characteristics are important for smart spaces.

In spite of these advantages, existing networking approaches fail to address the defining feature of smart spaces: smart spaces have *hundreds of heterogeneous, networked computers per person*. Smart spaces have hundreds of computers per person because each piece of equipment will be tagged, each object visible output will provide sensor data to the network, and each object with controls will be manipulatable from the network.

Too frequently, existing protocols require manual intervention to configure and to use. Smart spaces will have hundreds of often heterogeneous objects which frequently change configuration (as they move, are deployed, and wear out) and may be physically inaccessible. The implication of hundreds of nodes per person under these conditions is that failure to autoconfigure must be considered complete failure; failure of self-organization must be considered undeployability, and failure of automatic resource discovery must be considered effective inaccessibility. A corollary of large numbers of nodes is that smart spaces must employ algorithms which function

in the face of partial operation and heterogeneity.

Although we believe that coping with large numbers of nodes is the fundamental challenge behind smart spaces, two other issues are also important. As "spaces" suggests, smart spaces will be part of the physical world. This implies that they must understand their place (physical location and electronic neighborhood) and be able to influence other objects. Also smart-space nodes will often be resource limited because of constraints of size, weight, mobility, or cost.

The remainder of this paper explores these issues further. We examine the implications of large numbers of nodes for networking protocols, application design and operation, and security.

2. CHALLENGES IN NETWORK CONFIGURATION

If the fundamental change of smart spaces is a hundredfold increase in the number of networked computers per person, then this change is sure to have effects up and down the network stack.

At the physical layer, many nodes imply a need for very flexible network connectivity. Wireless technologies are the obvious choice here. Substantial progress has been made both in COTS wireless (for example standardization and widespread deployment of IEEE-802.11 [5]) and in the research community (for example, DARPA's GloMo efforts). We believe that flexible wired connections are also important. Wired connections can provide both high-speed connectivity and (if accompanied by power) can eliminate energy constraints.

At the network layer fully automated Internet address configuration is a requirement. IPv6 holds some promise here with a carefully considered autoconfiguration [11], but it assumes that failures can be manually resolved, it bases configuration on globally unique link-layer addresses, and its protocols require a very large address space. Approaches to relax these constraints are important if smart spaces are to apply to a range of link layers and to interoperate with IPv4 systems.

At the transport layer, smart spaces suggest wide use of multicast protocols [1, 6]. Multicast protocols allow groups of devices or a user and a group of sensors to interact efficiently. Multicast group addresses decouple the data sender and list of recipients. With thousands of nodes where weak connections and node failure may be common, flexible group membership is critical. Finally, announce/listen-style multicast-based protocols and soft-state simplify failure recovery.

Protocols which self-adapt to congestion are vital for smart spaces.

Floyd argues that it is the use of end-to-end congestion control and particularly TCP's additive-increase/multiplicative-decrease algorithms that have allowed the Internet to cope with orders of magnitude in growth [2]. If smart spaces are to adapt from the very high node densities of a conference room to a relatively sparse city street, similar protocols will be required.

3. CHALLENGES IN APPLICATION CONFIGURATION

Successful network autoconfiguration will allow nodes in smart spaces to talk with each other, and will allow them to do so without overwhelming the network. In addition, application-level approaches are required to make sense out of this communication.

Since smart spaces involve the physical world, absolute or relative physical location is an important concept. Node locations must be automatically configured; manual configuration will be both inaccurate and time consuming. GPS is an obvious candidate for location determination, but several factors argue the need of additional approaches: GPS accuracy is limited, it requires an antenna which may be too large for some nodes, reception is limited indoors, and cost remains a factor. Supplemental protocols are important to offset these problems. Accuracy of centimeters or tens of centimeters is important to place a node on one side of a wall or the other. The use of alternate protocols may be desired in buildings [14]. Finally, a location-equivalent of the Network Time Protocol [8] is needed to allow nodes to benefit from nearby GPS receivers.

Although physical location is important, we have argued elsewhere that *logical* location is often more important to real world applications [4]. (After all, do you care that you're at latitude 33.97988N, longitude 118.43994W, or that you're at work and that you're in your office?) Improving mappings between raw physical location and logical location are key to effective interaction between smart spaces and the environment.

Resource rendezvous is the process of matching clients and servers based on physical location and other attributes. Automated rendezvous allows applications to share data at high-levels without user involvement. Automated rendezvous includes both yellow-pages services (this light switch controls the northwest bay of lights) and more complex attribute-based queries (this light switch requests 25% illumination around the podium). Traditional approaches such as broadcast and expanding-ring search apply to resource discovery in smart spaces, but new opportunities exist to take advantage of physical location and device-to-device communication.

In addition to low-level congestion control, coordination at the application level is important to controlling network usage. We expect that devices in smart spaces will self-organize into *functional clusters*. Each cluster will be performing a higher-level coordinated action. For example, devices attached to individual lighting elements of a light panel may coordinate to dynamically vary their overall intensity based on environmental factors. Self-organization protocols build on the basic announce/listen protocols; announcements allow frequent cluster self-organization and reorganization. While existing clustering protocols [3] can be adapted to smart spaces, two features distinguish clustering in this environment:

- Clusters for smart spaces frequently overlap. Functionally related devices (e.g., lighting elements) may not be topologically contiguous, and different uses may overlap (full room lighting

vs. podium lighting). Classical clustering protocols usually focus on the formation of disjoint clusters.

- Some smart-space devices may be power-constrained. Moreover, these power-constrained devices may communicate with tethered devices that are not subject to such constraints. Clustering protocols will need to be aware of energy constraints, where possible, in order to limit communication.

A final implication of large numbers of devices is application heterogeneity. Smart spaces will be composed of multiple generations of smart nodes; applications must cope with many different protocol versions. Two very different approaches address this problem. On one hand we can construct nodes to be field-upgradable. Active Networks initiatives offer promise here [10]. An alternate approach is based on the principle of delay and discard. Nodes interact with very flexible protocols (for example, information buses [9]), delaying the need for frequent change. Nodes are designed to be cheap enough that they can simply be discarded when they no longer interoperate.

4. SECURITY CHALLENGES

Just as configuration must scale to support hundreds of nodes per person, so must security protocols. Existing protocols for key distribution are important, but more understanding is needed for ways to get good security for thousands of nodes instead of perfect security for tens of nodes. With many nodes, some will be compromised, so approaches to identify, isolate, and respond these nodes during continued operation are important [7, 13]. Finally, for very small nodes, new lightweight security protocols may be of increasing importance [12].

5. CONCLUSIONS

In this paper we have argued that the key problem facing use of smart spaces are the implications of configuring and using hundreds of computers per person. We have examined these issues for network communications, application interaction, and security implications. Addressing these problems are important if deployment of smart spaces is to become feasible.

Acknowledgements

The authors would like to thank the members of the Simple Systems ISAT study (chaired by Deborah Estrin) for their discussions on issues related to smart spaces.

This paper is also available as USC technical report number 98-677, July 1998.

References

1. Stephen E. Deering and David R. Cheriton. Multicast routing in datagram internetworks and extended lans. *ACM Transactions on Computer Systems*, 8(2):35-110, May 1990.
2. Sally Floyd and Kevin Fall. Promoting the use of end-to-end congestion control in the internet. Submitted to IEEE/ACM ToN, February 1998.
3. Ramesh Govindan, Cengiz Alaettinoğlu, and Deborah Estrin. Self-configuring active network monitoring (SCAN). White paper, February 1997.
4. John Heidemann and Dhaval Shah. Experiences with user-configurable, location-aware scheduling. Technical Report 98-675, University of Southern California, April 1998. submitted for publication.

5. IEEE. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (IEEE 802.11)*. IEEE, 1997.
6. Van Jacobson. Multimedia conferencing on the internet. Tutorial at SIGCOMM '94; published as a special issue of the SIGCOMM Newsletter, August 1994.
7. Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.
8. Dave L. Mills. Network time protocol (version 2) specification and implementation. RFC 1119, Internet Request For Comments, September 1989.
9. Brian Oki, Manfred Pfluegl, Alex Siegel, and Dale Skeen. The information bus—an architecture for extensible distributed systems. In *Proceedings of the 14th Symposium on Operating Systems Principles*, pages 58–68, Asheville, North Carolina, USC, December 1993. ACM.
10. David L. Tennenhouse, Jonathan M. Smith, W. David Sincoskie, David J. Wetherall, and Gary J. Minden. A survey of active network research. *IEEE Communications Magazine*, 35(1):80–86, January 1997.
11. S. Thomson and T. Narten. Ipv6 stateless address autoconfiguration. RFC 1971, Internet Request For Comments, August 1996.
12. Gene Tsudik and Brian Tung. On constructing optimal one-time signatures. Submitted for publication, November 1997.
13. Brian Tung. Crisis: Critical resource allocation and intrusion response for survivable information systems. <http://gost.isi.edu/projects/crisis/>, January 1998.
14. Andy Ward, Alan Jones, and Andy Hopper. A new location technique for the active office. *IEEE Personal Communications Magazine*, 4(5):8–15, October 1997.

Enabling “Smart Spaces:” Entity Description and User Interface Generation for a Heterogeneous Component-based Distributed System

Todd Hodes, Randy Katz

Computer Science Division
University of California, Berkeley
Berkeley, CA 94720-1776
{hodes,randy}@cs.berkeley.edu

July 17, 1998

Abstract

This paper motivates and describes a document-centric framework for component-based distributed systems. In the framework, XML documents are associated with programs that provide either static, immutable interface descriptions as advertisements of functionality at the server-side, or specification of manipulations of these server descriptions to express their usage at clients. We illustrate how the framework allows for 1) remapping of a portion of an existing user interface to a new room control (for example, due to movement of the terminal) 2) viewing of arbitrary subsets and combinations of the functionality available, and 3) mixing dynamically-generated user interfaces with existing user interfaces.

The use of a document-centric framework in addition to a conventional object-oriented programming language provides a number of key features. One of the most useful is that it exposes program/UI to referent objects mappings, thereby providing a standard location for manipulation of this indirection.

1 Introduction

Work in “smart spaces” overlaps a number of related research areas — active networking [1], networking middleware [2, 3], advanced user interfaces [4], mobile computing [5], and “ubiquitous computing” [6], to name a few — while at the same time focusing on a novel new domain for integration and extension of such work. One of the most challenging new issues is coping with the inherent *application heterogeneity* caused by the need for a location to adapt to individual users rather than vice-versa. Specifically, the challenge is allowing dynamic adaptation and reconfiguration of end-user applications in response to changes in available functionality and modes of user interaction. If users can only interact with “conventional” software, they will be constrained in their ability to customize the space to the extent in which the designers of the applications in it predicted their needs and/or provided a programmatic interface. Most often, the degree of freedom is limited by the user interface: application state and application preferences can only be manipulated through it, and the user interface itself is not highly configurable. Such is the case with monolithic programs, which are notoriously deficient in providing hooks into application state or in making it accessible via a well-defined external protocol. (On the other hand, monolithic user interfaces provide ease-of-use for well-understood functionality.) Client/server programs fare better, in that they provide (by definition) an exposed protocol, but suffer two problems: the specification of the interface is often ad hoc, and only a programmer can

make changes to their view of the functionality (by modifying the client code).

One approach to addressing these problems is to allow application programs to be downloaded on-the-fly to handheld devices and uploaded to local computers [7]; for example, as Java applets. The difficulty of this approach, though, is that it does not allow the end-user to customize applications for interaction with a heterogeneous set of services as related entities. In other words, it cannot overcome minor differences in protocol — even for functionally identical services — because the applications are opaque. For example, imagine needing to download a different application upon encountering every new light switch in the world in order to use it. Though the functionality is exposed, it is not in a form amenable to manipulation.

Another approach is to standardize functional interfaces and require that applications (and spaces) support these standards, thereby avoiding the above bullet. The difficulty here, though, is that enforcing a plethora of application-specific standards is impractical.

Clearly, a different model than either of the above is preferable, one that balances the need to expose interfaces with the need to agree on protocol standards. Herein, we propose such a new approach, leveraging a component-based application framework [8] and pairing it with an architecturally-independent document model for component descriptions. It hybridizes features of the two basic approaches discussed above, allowing downloading/uploading of code fragments (as specified by the documents) and imposing a standard for interface description and manipulation that is not application-specific.

This paper describes our initial thoughts and work on this infrastructure, a novel prototype of a document-centric framework for component-based distributed systems. In the framework, application programs and user interface programs are associated with documents that provide one of either

- static, immutable interface descriptions as advertisements of functionality, or
- specification of manipulations of these server descriptions to express their usage.

We describe the framework and illustrate how it allows for

- remapping of a portion of an existing user interface to a new room control (for example, due to movement of the terminal)
- viewing of arbitrary subsets and combinations of the functionality available, and
- mixing dynamically-generated user interfaces with existing user interfaces.

The use of a document-centric framework in addition to a conventional object-oriented programming language provides a number of key features, the most critical is that it exposes program/UI to referent objects mappings, thereby providing a standard location for users (and their programs) to manipulate these mappings.

The rest of this paper is structured as follows. Section 2 introduces our document-centric component framework design. Section 3 describes the investigation approach. Section 4 introduces XML and why we choose it as our syntax. Section 5 describes the markup (tags) used in the documents and how they are used in automatic user interface generation. Sections 7–8 give examples of the software usage in example applications and show document markup examples along with their related user interfaces. Section 9 describes continuing work, and, finally, Section 10 summarizes and concludes.

2 Component Frameworks and the Document-centric Approach

Component- and object-based middleware platforms for large-scale heterogeneous environments are appearing at the fore of distributed systems research. Such systems provide support for object instantiation, discovery, naming, and communication based upon remote method invocation mechanisms (c.f., CORBA [9] or Enterprise Java Beans [10]). Continued evolution of such systems provides for a wide-ranging set of extensions to the basic model. Example directions include supporting 1) availability, fault-tolerance, and persistence to object stores [11], 2) uniform, language-independent “wrapping” around arbitrary data producers and consumers to make them look like objects to the system [12], 3) group communication primitives (i.e., tuple-spaces [13], reliable multicast [14], queued event notification [15]), and 4) system facilities for *dynamic, ad hoc* creation or extension of *end-user applications* from a set of constituent distributed components.

Designing systems with such a plethora of components is difficult even if it is simply considered a massive integration effort (i.e., even though many of the constituents exist); the more interesting and challenging design issue is that of determining the syntax and semantics used to describe components and how they are used. This defines the programming model. A traditional approach is to expose distributed components as objects that can be manipulated naturally from within an object-oriented programming language, albeit in a restricted fashion if the remote object isn’t native. Actions are applied to data residing inside running applications using only well-defined library APIs to manipulate system state. On the other hand, a different approach is to externalize a portion of the system state in the form of *documents* that describe components and interactions. This allows state to be manipulated by *editing documents* in addition to allowing for access via an API — basically, *allowing system manipulation via authoring in addition to programming*. We call this relatively novel second approach a *document-centric* approach, and will focus on it for the rest of this paper.

Any Turing-complete language paired with a network can be used to encode the specification of a program built from distributed components — regardless of whether it is object-oriented, imperative, functional, or whatever. Thus, a “document-centric” model does not provide additional functionality (at least not fundamentally), but instead only adds an additional layer of indirection. Documents are written using a declarative style and used in addition to application code. Their function is to focus on the specification of data and elide (or greatly reduce the amount of) control flow information. Using a document-centric framework to expose this control/data

separation (i.e., where documents are first-class entities rather than productions) provides a number of advantages, including

- aligning with the needs of cross-enterprise data-sharing, where the right thing to standardize is documents rather than APIs [16]
- simplifying inter-object usage specification, separating it from the programming chore and/or use of a particular monolithic application (manipulating documents to affect change is often simpler than editing and rebuilding programs, and less opaque than figuring out how to do through an non-standardized application user interfaces)
- providing for syntactic fault-tolerance ala HTML
- allowing clean incorporation of metadata, which can simply be directly added to a document rather than added as a new method or instance variable in an object
- providing for application-independent storage and manipulation — e.g., to maintain consistent per-object preference propagation, or to provide a single, concrete point for updates to a user’s environment

This disassociation of programs/UIs from the objects they reference is similar to the Model/View/Controller architecture from Smalltalk [17]. In the M/V/C architecture, data (the model) is separated from the presentation of the data (the view) and events that manipulate the data (the controller). Similarly, documents in our system act as the glue that associates data to user interfaces/programs that manipulate and view that data. This strict delineation 1) provides client device independence, 2) provides program/UI language independence, 3) exposes program/UI to referent objects mappings: they become explicitly manipulable, 4) makes explicit what objects to manipulate and how they can be manipulated, and 5) can be used to generate user interfaces when custom ones are not available or unacceptable.

This document-centric distributed object management framework is illustrated in Figure 1.

This project investigates using a document-centric framework for specifying interaction with and between a distributed set of “services” available over a wide-area internetwork¹, for example, the set of services made available in a series of “smart spaces.” It focuses on two aspects: description of the available services and flexible association of user interfaces to these services. We limit the scope of the discussion to a single (varying) collection of objects being referenced by a single (varying) user interface. This limits the problem domain so that we do not have to specify how the markup indicates paths of object-to-object interaction. Instead, we only have to describe interactions between the “endpoints:” users manipulating widgets to interact with a set of independent services. The focus, then, is on these two endpoints.

We proceed from the following observations:

- Services require concrete, immutable interface descriptions.
- Clients (or system proxies) need to manipulate and save collections of these interfaces.
- We would like a single document format that provides both functions.

From these observations, we establish our design. A single document format is shared among all entities in the system. At the “server-side,” documents act as static interface definitions, similar to the use of IDL in CORBA. Elsewhere, documents act as a stable

¹We call application components that use our framework “services” to contrast with the more generic term “objects.”

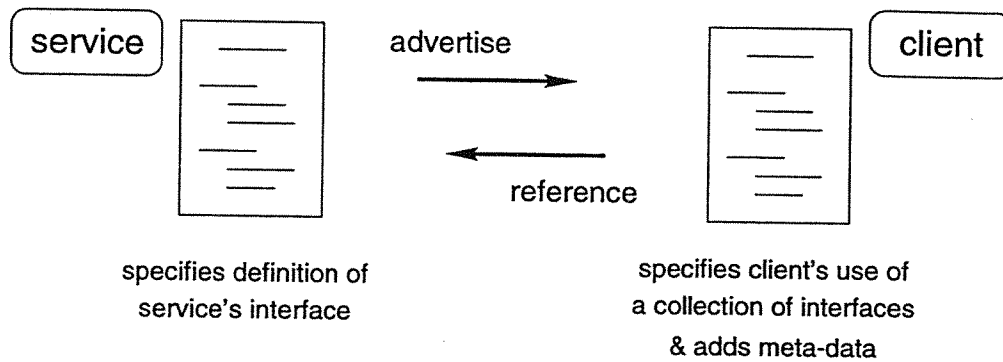


Figure 1: The Document-centric Model: Services are described by immutable, static documents that advertise the definition of their interface (ala the CORBA IDL or a Java interface); Clients maintain documents that indicate how the a collection of interfaces are used and maintain metadata about the collection

but manipulable (composable/decomposable) format for specifying object collections and references, defining interactions between objects in a collection, defining the object interfaces expected by programs and user interfaces, and storing arbitrary metadata about referents.

Specifically, there are two challenges. The first challenge is defining a single document schema that:

- notates services' available functionality, or *interface*,
- associates relevant programs and UIs to collections of services, or, vice-versa, lists the service interfaces expected by particular programs
- can flexibly compose and decompose based on constituent elements, and
- allows for easy incorporation of service-specific meta-data (i.e., without affecting existing functionality that does not expect it and in a self-describing manner).

The second challenge is providing software that can use documents written in the schema to generate user interfaces when custom ones are not available, mixes custom and generated user interfaces as necessary, and implements the run-time environment.

3 Project Approach

Our approach to the problem is threefold. Leveraging the eXtensible Markup Language (XML) [18] for syntax, we develop our schema as an XML document type definition (DTD). The schema provides markup tags for language-independent service descriptions and for mapping UIs (programs) to referent objects and vice-versa. We then build software that can heuristically generate UIs from these service descriptions without associated custom UIs, and allows mix-and-match use of custom and generated UIs. Additionally, we built an index application that lists the collection of available UIs and objects, allowing a combination of them to be interactively selected for presentation on the user's machine. Finally, we prototype applications that use the model, manually construction and editing documents to simulate how programs would automatically manipulate them.

For our prototype application domain, we focus on a set of location-based services [7] that provide software remote control

of room devices from a wirelessly-connected laptop computer. Manipulations of the applications' documents allows the controls to adapt as the environment changes around the user. Specifically, the manipulations provide for

- the remapping of a portion of an existing user interface to a new room control (for example, due to movement of the terminal)
- viewing of arbitrary subsets and combinations of the functionality available in the space, and
- mixing generated user interfaces with custom user interfaces

This functionality is easily represented as operations on documents containing the associations between between programs/UIs from the objects they reference, exactly the model as described above.

4 XML

We have chosen to build atop the extensible markup language (XML) for our schema design, leveraging its allowances for the creation of custom, application-specific markup languages.

XML is an SGML subset providing self-describing custom markup in the form of hierarchical named-values and advanced facilities for referencing other documents (ala the HTML `<href>` tag). It is one protocol among a group that is touted as the successors to HTML (The companion protocols are XSL for style sheets and XLL for new linking mechanisms). XML includes ability to specify, discover, and combine a group of associated document schemata — otherwise known as *document type definitions* or DTDs. Examples include a growing set of metadata markup proposals such as Resource Description Format, the Dublin Core, XML-Data, and others.

Unlike HTML, the set of tags in XML is flexible; the tag semantics are defined by a document's associated DTDs. A key property of XML, then, is that it is *dependent* on these schema to be useful, and dependent on agreements in schema to allow interoperability. Thus, the problem is defining schema syntax (the tagset and their relationship) and agreeing on how a schema's associated "browsers" (borrowing the HTML term) semantically interpret these tags.

We believe there is a natural synergy between XML's need for schemata and the specification requirements of distributed object systems — the former provides a self-describing and extensible

syntax with a rapidly expanding set of metadata tags; the latter provides a programming model for “web objects” described in XML.

5 XML Tags for Object Description

We use six tags in our initial design. Other tags that appear in our documents are assumed to be application-specific metadata, and can be ignored by programs that do not understand them. We now describe each tag in turn. The actual DTD specification is provided in the Appendix A.

The `<object>` tag is a container tag. It has one optional attribute, “name”, which is either a string or reference identifying the type/class of the object interface being described. It can contain at most a single `<label>` tag, zero or one `<addrspec>` tags, any number of `<ui>` tags, and any number of `<method>` tags. When converted to a user interface, an `<object>` is instantiated as a frame (a container for other widgets).

The `<label>` tag provides a text description of the object which contains it. It has no optional attributes. It can contain no additional internal tags except those providing text formatting. When converted to a user interface, the `<label>` tag is used as a title for its parent object’s frame.

The `<addrspec>` (address specification) tag indicates the address and port number on which its parent object listens for method invocations and events. An uninstantiated object will have no `addrspec` tag. It can contain no additional internal tags and does not have any optional attributes. When converted to a user interface, the `<addrspec>` tag is used as the location to which any method calls are sent (currently via string-based UDP messages).

The `<method>` tag defines the name of a method that can be invoked on the object in which it is contained. It has two optional attributes: “name”, which is name of the method call, and “lexType”, which indicated the lexical type of messages returned due to the method call (the list of lexical types is described below). The `<method>` tag can include (only) zero or more `<param>` tags. When used in automatic user interface generation, each `<method>` tag is mapped to a frame with contents. The name of the method is placed on a button at the top of this frame; pressing this button invokes the method call. At the bottom of the method frame, a label is appended for textual representation of replies. Note that in our system, method invocations and returns are asynchronous, event-based messages (like active messages [19]) rather than blocking remote procedure calls. Thus, update events (“replies”) can actually occur at any time, independent of the manual invocations at the client. In this manner, `<method>` tags can also be used as a means for subscribing to updates from the object.

The `<param>` tag indicates a parameter to the `<method>` tag that encloses it. It has two optional attributes, “lexType”, indicating the lexical type of the parameter, and “optional”, a boolean tag that indicates whether the parameter is required or optional. The `<param>` tag may have no additional internal tags, and its contents are assumed to be the name of the parameter. For UI generation, parameters are mapped to individual user interface widget objects. Each of these widget objects support a `get_val` method that returns the current widget setting. It is used to marshal the parameters for method invocations. The mapping from lexical type to UI widgets is described in Section 6.

The `<ui>` tag is used to associate a particular program to the object in which it is specified. The contents of the tag is assumed either to be a string indicating the name of an existing user interface object that will reference the document object description (assumed to be known or discoverable out-of-band), or the address and port number where such a user interface object can be requested. Only

the former is currently implemented. It has one possible attribute, “lang”, indicating the language of the indicated program.

In our framework, documents are used in addition to application code, not instead of it. The documents act to specify what programs are needed and how they are run (via the use of `<ui>` tags at various levels in a hierarchical description of a service). The assumption is that the indicated applications will reference the documents, respecting the indirection exposed by the document.

6 Automatic User Interface Generation

Many of the mechanics of generating user interfaces from interface descriptions were described in the preceding section. The remaining features to be discussed are the heuristic mapping from lexical types to user interface widgets, and how custom user interfaces indicated by a `<ui>` tag can be intermingled with these custom user interfaces.

We currently have implemented mappings from lexical types to objects wrapped around Tk [20] UI widgets in the mash shell [3]. Permissible lexical types include int, real, boolean, enum, string. The int and real type can have an optional range modifier. They are mapped to widgets as follows: an int or real with a “range” modifier is mapped to a scale widget (a slider). Without a range modifier, they are mapped to an entry widget (a type-in box). A boolean is mapped to a check-button widget (a toggle switch). An enum is mapped to a list of radio-buttons (one-of-N list selection). A string is mapped to an entry widget.

As for co-mingling these generated collections and existing UIs referenced in `<ui>` tags, the granularity of reference is at the level of individual objects. Thus, all objects receive a frame, and it is filled with either the custom-generated contents mapped from `<method>` and `<param>` tags, or the existing UI. The latter is handed a window handle and is expected to instantiate itself as a child of that window handle.

7 The Framework in Action: Examples

We now illustrate some examples. Each highlights a different element of the design of the overall architecture.

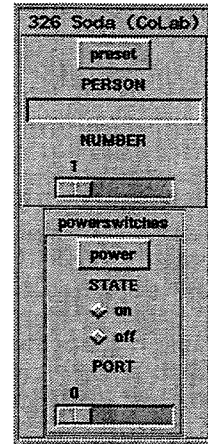
The first example shows an XML document that describes the interface to a portion of the functionality available in Soda Hall’s “CoLab” (Collaboration Laboratory, borrowing Xerox PARC’s terminology) and the resulting automatically generated user interface to it, as shown in Figure 2. The document describes two objects, one contained in the other. The outer object implements a method for setting a preset for the entire room; the inner object is one of the objects referenced by the outer object (i.e., one of the things set by the preset) and implements its own interface independently — an interface to a pair of power switches in the room. The `<param>` tags contain various lexical types, illustrating our use of heuristic mapping to widgets. This utility of this basic functionality is that it allows users the possibility to interact with dynamically discovered objects, not just programmatically, but also directly.

The second example illustrates the combination of a custom user interface with a generated one. The document is identical to that in the previous example except a single new tag is added: a `<ui>` tag to the internal (power switch) object, as shown in Figure 3(a). This causes that object’s interface to be replaced by the UI object referenced in the tag rather than generated on-the-fly. The resulting difference is illustrated in Figure 3(b). This example illustrates how dynamic extensions to existing applications can be seamlessly

```

<object name="326">
  <label> Soda (CoLab) </label>
  <addrspec>spade.cs.berkeley.edu/0001</addrspec>
  <method name='preset'>
    <param lextype="string"> person </param>
    <param lextype="int:range=1-8"> number </param>
  </method>
  <object name='powerswitches'>
    <label>powerswitches</label>
    <addrspec>spade.cs.berkeley.edu/0002</addrspec>
    <method name='power'>
      <param lextype="enum:on,off"> state </param>
      <param lextype="int:range=0-1"> port </param>
    </method>
  </object>
</object>

```



(a) XML document

(b) User interface

Figure 2: An example document and generated user interface.

incorporated using our architecture, a form of “plug-in” architecture similar to that used in, say, Photoshop or with Visual Basic extensions.

The third example illustrates use of the indirection exposed by the use of the “document-centric” model by replacing a referent under a multi-object <ui> tag. The document fragment shown in Figure 4(a) is assumed to be used by an existing application. The user interface for this application is a custom-designed monolithic interface referenced in the topmost <ui> tag. In Figure 4(b), one of the component objects in the container object has been replaced. Because the type of the referenced object remains the same, only the <addrspec> tag changes. The result of this change is that the application looks the same, but a portion of it now references a new service. This function illustrates the possibility for remapping interfaces due to, e.g., terminal mobility or fault tolerance. Specifically, the example takes a portion of the document describing the interface to the 405 Soda Hall seminar room and remaps the light switch portion of the interface to a new switch.

The fourth and final example illustrates the ability to use a subset of presented functionality. The document in Figure 5(a) is the same as that from Figure 4, except all the internal objects referenced from the outermost container object have been ripped out. The resulting user interface is presented in Figure 5(b). This example shows how a user can easily elide material not considered relevant or not frequently used. In this case, we leave only the interface to the light switch exposed, simulating the case where the user has chosen to save screen real estate because, e.g., controlling only the lights is the most common usage.

8 Indices and the End-user Environment

In addition to building software to parse the particular XML DTD and spit out interfaces, we need to provide the user with a way to manage the set of documents and available interfaces. We provide this functionality through use an “index” application, shown in Figure 6. One the left side of the application, all objects are listed by type and address specification. Each type has an associated

document and an associated user interface. When one of the check-buttons beside an object name is set, the associated user interface is displayed for use by the user.

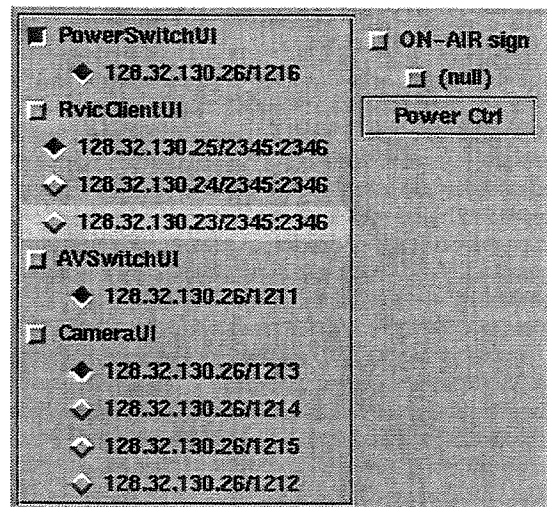


Figure 6: The Index application lists all available interfaces and allow the user to interactively select which ones he or she wishes to use. Illustrated here, the user has selected the user interface to the power switches in the Berkeley CoLab.

9 Continuing Work

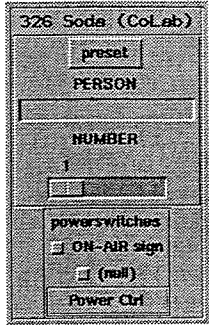
Another set of applications that would be useful for the end-user are those that would allow easily manipulation of documents themselves rather than simply the results of having documents. Design and implementation of this piece is ongoing.

```

<object name="326">
  <label> Soda (CoLab) </label>
  <addrspec>spade.cs.berkeley.edu/0001</addrspec>
  <method name='preset'>
    <param lextype="string"> person </param>
    <param lextype="int:range=1-8"> number </param>
  </method>
  <object name='powerswitches'>
    <label>powerswitches</label>
    <addrspec>spade.cs.berkeley.edu/0002</addrspec>
    <ui lang=mash> PowerSwitchUI </addrspec>
    <method name='power'>
      <param lextype="enum:on,off"> state </param>
      <param lextype="int:range=0-1"> port </param>
    </method>
  </object>
</object>

```

(a) XML document



(b) User interface

Figure 3: An example document and associated user interface, this time where a <ui> tag allows for the incorporation of a custom UI in addition to the generated components.

Currently, the <ui> tag can only reference local objects. We are in the process of implementing the remote retrieval protocol.

A logical next step of this work is dealing with mismatched types. For example, assume a light switch in some locale implements a different interface than the one in the user's home environment. Rather than require the use of a dynamically-generated user interface, we'd prefer to allow for the use of an existing user-interface. To do so, we must transparently remap method invocations to the new location and also remap the call parameters to match the new type. Incorporating such functionality allows far more flexibility in the reuse of existing user interfaces and intermingling of existing interfaces and discovered objects. The price is that it requires the use of external transformational operators that provide type coercion for method calls. Fortunately, such transformational operators could be written once, reused, and shared among the community of users; additionally, they could be chained together in order to provide new type-to-type coercions. This functionality is a natural extension of our framework. The difficulty of this approach is not in creating these mapping operators and storing them in a shared repository, but instead that of building the use of them into the end-user software. Users should be able to visually manipulate object mappings and the correct transformations should be done automatically. As a concrete example, this means that when a new light switch is discovered, the user should be able to indicate which program element should manipulate it, and any required remapping of method calls — i.e., document manipulations — should be done automatically, though possibly heuristically. Additionally, users should then be able to easily modify these mappings. Currently, this is all done via manual manipulation of documents rather than automatically.

Another important extension of this work is designing how to notate one object's use of other objects so as to allow for, e.g., multiple interfaces. The transformational operators described above are simple examples of such objects, in that they require separation of input and output interface descriptions. This requires extension or modification of our schema.

Finally, in order to allow for programmers to more easily use this document-centric model — without having to manually create

interface description documents — we would like to automatically generate the documents from Java objects and other distributed component system pieces. To do so, we can leverage the CORBA Interface Definition Language (IDL) [9], for which there are mappings to C, Java, Ada, and other languages. We can integrate components written in these languages by creating a mapping from IDL descriptions to our XML schema and implementing it. Our object interface schema will need to be extended to support structured types in order to allow such a mapping.

10 Summary

We have described a document-centric framework for description and interaction with entities in a distributed object system. We have shown how the framework allows for:

- the remapping of a portion of an existing user interface to a new room control (for example, due to movement of the terminal)
- viewing of arbitrary subsets and combinations of the functionality available in a “plug-in”-style architecture, and
- mixing dynamically-generated user interfaces with existing user interfaces.

The use of a document-centric framework in addition to a conventional object-oriented programming language

- provides client device independence,
- provides program/UI language independence,
- exposes program/UI to referent objects mappings: they become explicitly manipulable,
- makes explicit what objects to manipulate and how they can be manipulated, and
- can be used to generate user interfaces when custom ones are not available or unacceptable.


```

<object name="405">
  <label> 405 Soda (HTSR) </label>
  <addrspec>htsr.cs.berkeley.edu/0000</addrspec>
  <ui lang='tcl/tk'>htsr.cs.berkeley.edu/6903</ui>
  <object name='lights'>
    <label>lights</label>
    <addrspec>htsr.cs.berkeley.edu/6902</addrspec>
    <method name='power'>
      <param lextype="enum:on,off,dim"> state </param>
    </method>
  </object>
  <object name='vcr'>
    ...
  </object>
  ...
</object>

```

(a) Original XML document

```

<object name="405">
  <label> 405 Soda (HTSR) </label>
  <addrspec>htsr.cs.berkeley.edu/0000</addrspec>
  <ui lang='tcl/tk'>htsr.cs.berkeley.edu/6903</ui>
  <object name='lights'>
    <label>lights</label>
    <addrspec> 205south.sims.berkeley.edu/9999 </addrspec>
    <method name='power'>
      <param lextype="enum:on,off,dim"> state </param>
    </method>
  </object>
  <object name='vcr'>
    ...
  </object>
  ...
</object>

```

(b) Document with replaced referent

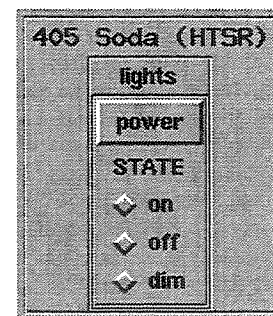
Figure 4: Remapping of function by replacing a referent under a multi-object <ui> tag. A fragment of the "original" document is shown in (a); the modified document is shown in (b), where the only difference is the new <addrspec> tag. (The <addrspec> tags are highlighted.)

```

<object name="405">
  <label> 405 Soda (HTSR) </label>
  <addrspec>htsr.cs.berkeley.edu/0000</addrspec>
  <object name='lights'>
    <label>lights</label>
    <addrspec>htsr.cs.berkeley.edu/0000</addrspec>
    <method name='power'>
      <param lextype="enum:on,off,dim"> state </param>
    </method>
  </object>
</object>

```

(a) XML document



(b) User interface

Figure 5: Subsetting functionality. The example illustrates how functionality can be aggregated or subsetting by modifying the document associated with a program. The full description of the interface to 405 Soda has been cut down so that only a single object remains. The user interface is updated accordingly.

To implement our scheme, we designed a XML schema and accompanying software that

- notates services' available functionality, or *interface*,
- associates relevant programs and UIs to collections of services, or, vice-versa, lists the service interfaces expected by particular programs
- can flexibly compose and decompose based on constituent elements, and
- allows for easy incorporation of service-specific meta-data (i.e., without affecting existing functionality that does not expect it) via the self-describing nature of XML.

Using a peer document or "description" alongside an application to provide for much of the flexibility described here has been described before [21], but only in the abstract. It is the advent and growing popularity of XML and distributed object programming that synergistically combine to give us a syntax for these descriptions and a concrete framework for their use.

A Schema DTD

The document type definition for the XML files used by our system is as follows:

```
<!ELEMENT object (label?, addrspec?, ui*,
                 method*, object*)>
<!ATTLIST object
  name CDATA #REQUIRED>
<!ELEMENT method (param*)>
<!ATTLIST method
  name CDATA #REQUIRED>
<!ELEMENT param (#PCDATA)>
<!ATTLIST param
  name CDATA #REQUIRED
  lexType (int | real | boolean | enum
           | string | ...) 'string'
  optional #BOOLEAN>
<!ELEMENT label (#PCDATA)>
<!ELEMENT addrspec (#PCDATA)>
<!ELEMENT ui (#PCDATA)>
```

References

- [1] D. Tennenhouse, J. Smith, W. Sincoskie, D. Wetherall, and G. Minden. A Survey of Active Network Research. *IEEE Communications Magazine*, pages 80–86, January 1997.
- [2] A. Fox, E. Brewer, S. Gribble, and E. Amir. Adapting to Network and Client Variability via On-Demand Dynamic Transcoding. *ASPLOS*, 1996.
- [3] Steven McCanne, Eric Brewer, Randy Katz, Lawrence Rowe, Elan Amir, Yatin Chawathe, Alan Coopersmith, Ketan Mayer-Patel, Suchitra Raman, Angela Schuett, David Simpson, Andrew Swan, Teck-Lee Tung, David Wu, and Brian Smith. Toward a Common Infrastructure for Multimedia-Networking Middleware. *Proc. 7th Intl. Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '97)*, May 1997.
- [4] B. MacIntyre and S. Feiner. Future Multimedia User Interfaces. *Multimedia Systems Journal*, 4(5):250–268, October 1996.
- [5] Brewer, Katz, et al. *A Network Architecture for Heterogeneous Mobile Computing*. submitted for publication, IEEE Personal Communications.
- [6] M. Weiser. Some Computer Science Issues in Ubiquitous Computing. *Communication of the ACM*, 36(7), July 1993.
- [7] Todd Hodes, Randy Katz, E. Servan-Schreiber, and Larry Rowe. Composable Ad hoc Mobile Services for Universal Interaction. *Proceedings of the 3rd ACM International Conference on Mobile Computing and Networking*, pages 1–12, 1997.
- [8] David Krieger and Richard Adler. The Emergence of Distributed Component Platforms. *IEEE Computer Magazine*, pages 43–53, March 1998.
- [9] Object Management Group. Common Object Request Broker Architecture. <http://www.omg.org/>.
- [10] Sun Microsystems. Enterprise Java Beans. <http://java.sun.com/ejb>.
- [11] J. Eliot, B. Moss, and Tony L. Hosking. Approaches to Adding Persistence to Java. *First International Workshop on Persistence and Java*, September 1996.
- [12] Charles Axel Allen. Automating the Web with WIDL. *World Wide Web Journal*, 2, 1997.
- [13] IBM Almaden Research. TSpaces. <http://www.almaden.ibm.com/cs/TSpaces>.
- [14] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang. A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing. *ACM SIGCOMM 95*, pages 342–356, August 1995.
- [15] A. Joseph, A. deLespinasse, J. Tauber, D. Gifford, and M. Frans Kaashoek. Rover: A Toolkit for Mobile Information Access. *Proceedings of the Fifteenth Symposium on Operating System Principles*, December 1995.
- [16] Robert Glushko. The XML Revolution. UC Berkeley SIMS Symposium Presentation, April 1998.
- [17] G. Krasner and S. T. Pope. A Cookbook for Using the Model View Controller User Interface Paradigm in Smalltalk-80. *Journal of Object-Oriented Programming*, August/September 1988.
- [18] World Wide Web Consortium. eXtensible Markup Language. <http://w3c.org/XML/>.
- [19] T. von Eicken, D. Culler, S. Goldstein, and K. Schauer. Active Messages: a Mechanism for Integrated Communication and Computation. *International Symposium on Computer Architecture*, May 1992.
- [20] J. K. Ousterhout. *Tcl and the Tk Toolkit*. Addison-Wesley Publishing Company, Reading, MA, 1994.
- [21] T. Hodes and R. Katz. Composable Ad hoc Location-based Services for Heterogeneous Mobile Clients. *ACM Wireless Networks*, 1998. Special issue on Mobile Computing, to appear.

Smart Paper: Techniques for Hybrid Paper Electronic Interfaces

Scott Hudson

HCI Institute, School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
{hudson@cs.cmu.edu}

1. INTRODUCTION

Paper is a well-evolved media for expressing and recording information. It is ubiquitous, highly portable, and easy to use in a wide range of environments. Paper is inexpensive, can be annotated easily, and provides excellent readability properties. Further, paper documents offer excellent ergonomic properties. New display technologies have recently begun to approach the resolution and contrast properties of typical laser printed documents (e.g., the commercially available 300 dpi LCD display by dpiX inc.[1]). However, no current electronic displays can approach the economy, convenience, “feel”, and ease of use of, for example, a paperback book.

On the other hand, as we all know electronic presentation technologies have many advantages. They allow us to create and publish great quantities of information, and can make that information available worldwide in literally a matter of seconds. Further, electronic materials offer considerably more powerful capabilities such as searching, dynamic multimedia content, hyperlinking, and easy reuse. Finally, going to an electronic medium opens the door to not just information, but to computation — a fundamentally more powerful capability.

In order to combine the good properties of paper and electronic media, we propose that new technologies can be created which will allow computational activities to be combined with, or applied to, materials on paper. If some of the advantages of electronic materials can be added to paper materials, it will be possible to create new hybrid paper electronic systems that offer the best of both worlds. Such systems will be able to approach the very high level of convenience and ease of use of paper, while allowing access to the tremendous power of computational media. We believe that recent increases in computational power have now made such systems possible by providing very powerful computational elements (for example, at a level of power needed for computer vision problems) in small portable form factors.

2. APPLICATIONS


As an example of the type of interactions that we hope to be able to support, consider an electronically augmented textbook or manual. The printed paper form of this book would be placed in a computing device resembling a thick 3 ring binder. This device would contain a processor and one of several possible display and input devices — such as those outlined below — for augmenting the printing already on the paper. Alternatives for paper input and output will include techniques acting directly on the paper, techniques working over the paper (using a transparent overlay), and/or techniques implemented near the paper using a separate display area. The printed book would either be accompanied by supporting electronic information (much as many computer books currently come with programs and data on a CD-ROM), or preferably, simply contain a machine readable URL for retrieving augmentation information dynamically over the net.

In this setting, the main character of the book would be preserved. It could still be treated for the most part as a simple book and read in a conventional fashion — retaining the portability, readability, “feel”, and ease of use properties of a conventional book, with only a small electronic “binding” added. However with computational augmentation, the book could offer a number of new functionalities. As a simple example, to directly access an index, cross-reference, or glossary entry, the user might simply circle a word or phrase, then tap a small icon printed in the bottom margin of the book (as illustrated on this page). The relevant information could then be displayed in context — again using display alternatives such as: projection directly on the paper, or display via a transparent overlay.

Using these same basic capabilities of referring to (or *picking up*) content from the printed page, and indicating an action by pointing or taping, the user could also be given access to more sophisticated capabilities. For example, in an assembly or maintenance manual, it might be possible to augment a discussion with a small animation showing the exact procedure used for a

Index **Glossary**

Table 1: An Initial Capability Taxonomy for Smart Paper Interfaces	
Content:	Arbitrary / Known / Controlled
Input:	
Where:	On / Over / Near the Paper
What:	Input from the User Points, Command Selection, Strokes, "Other Values"
	Input from the Paper Object Selection, Text Data, Encoded Data, Recognition of Previous User Strokes
Other:	User Marks Permanent / Temporary Page Number Know / Unknown Page Position Known / Unknown
Output:	
Where:	On / Over / Near the Paper
Registration:	Fine-grained, Course-grained, None

particular model. For fixed presentations, the user could bring up this animation on demand by tapping on a small marker printed directly on the page (such as: ). More importantly, for procedures that varied based on hardware options, the user might pick from a printed set of menus listing known models, variations, and options in order to call up a customized presentation.

With only slightly more sophisticated input methods, it would be possible to make use of more sophisticated augmentations. For example, some dynamic displays might be derived from computational simulations. In that case, it may be very helpful to give the user control of several key simulation parameters via a slider-like interaction technique. Affordances for this technique might be printed directly on the page (as illustrated to the right) with actions carried out via taps or movements over the printing. As another example, a table of numbers, could be made available to be *picked up* and placed in cells of a spreadsheet (which might be displayed within an area of pre-printed cells on the paper and might include a printed set of buttons for composing common formulas). This would allow the user to perform exploratory, *what-if* type experiments to fully understand the concepts being presented. Finally, by embedding URLs (or small marks that encode a URL) it will also be possible to provide access to live or changing data from the web. So for example, a finance textbook could include not only fixed examples, but also examples drawn from today's actual interest rates or stock prices, as well as those from any selected historical period.

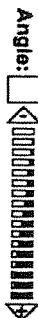
In general, by adding a few basic interaction techniques, we can allow the user to much more directly experience and experiment with the concepts in printed material, and turn it from a passive to an interactive media. While the specific functionality to be added to paper media is not itself new, we believe offering these capabilities in the context of the widely used, convenient, and conventional medium of paper, provides very

important usability benefits. Overall this work represents a significant step toward the general goal of adapting computers to human tasks and behaviors (rather than forcing humans to adapting to computers).

3. TECHNICAL CHALLENGES AND DESIGN APPROACHES

Technical challenges for this work come in two central areas. The first is the creation of devices that can provide input and output capabilities for paper-based media. Several specific alternative device concepts for providing the required capabilities will be described in the next section. Once these devices have been constructed, the second central technical challenge will be the creation of interaction techniques that smoothly integrate the paper and computational worlds.

To illustrate approaches to supporting input from, and output to, paper media, this section will present several proposed alternative device concepts. However, before considering specific devices concepts it is useful to consider a taxonomy of capabilities for smart paper systems. Table 1 shows such as taxonomy. The taxonomy has 3 major categories of capabilities: content knowledge, input aspects, and output aspects. Content knowledge indicates what the system knows about the content being portrayed on paper. Alternatives in this dimension include arbitrary, known, and controlled content. Arbitrary content would come from "external" sources such as a conventional book in a library. Known content assumes that the system knows the full contents of the material but did not print it. Finally, controlled content assumes that the system controlled the printing of paper content (and hence could embed special marks, control affordances, etc.). Arbitrary content offers the most ability to work with "natural" and real-world material, however, it also presents more limited opportunities for interaction (for example, "picking up"



content from paper is much easier if the content is known in advance). Known and controlled content represent less “natural” domains, but also provide increasingly more opportunities for integrating computation with paper.

Input in the taxonomy is broken into aspects of where, what, and “other aspects”, while kinds of input (“what” inputs) are categorized into information taken from the paper, and information provided by the user. Finally, output aspects of the taxonomy consider where the output occurs (*on*, *over*, or *near* the paper) and the level of registration that is required between paper and electronic presentations. (It is important to note that the dimensions of this taxonomy are not entirely independent, nor do all combinations of capabilities necessarily make sense together. For example, the notion of fine-grained output registration does not apply in systems using output near the paper.)

A first example of a device for interacting with real paper is shown in early prototype form in Figure 1 and described in [2] (with a later prototype being developed by Hitachi Research shown in Figure 2). This device is an augmented highlighting pen. It provides input directly from the paper (of previous user strokes and limited amounts of arbitrary printed content), and output near the paper. The device consists of a conventional highlighting pen with a miniature camera and tip switch attached. Using simple vision techniques, it is possible to recognize highlight marks previously made by the user on the paper. By associating electronic content with these marks, it is possible to give them meaning and significance in the electronic world. In particular it is possible to use these on-paper marks as hyperlinks from paper to electronic content. Once a mark has been made and a hyperlink association established, that hyperlink can be followed simply by pointing at it with the pen.

This prototype, while supporting only limited capabilities, has been useful to explore concepts for smart paper interfaces. Of particular interest with this device concept has been the ability to support dynamic “user constructed” interfaces. These interfaces consist of words or other marks on paper which represent commands. These marks are created by the user as they are needed using their own vocabulary (and simply associated with computational commands on the electronic side). This provides an interesting new style of very lightweight interaction, which is free form in nature, and hence matches the informality and flexibility normally associated with paper.

A second proposed device concept is designed to allow exploration of interaction techniques that occur over the paper involving known or controlled content. It will consist of a transparent LCD overlay display with a pen sensitive surface as illustrated by the drawing in Figure 3. This device can be viewed as a physical embodiment of the tool glass and magic lens interaction techniques [3]. With this device, the user acts on the

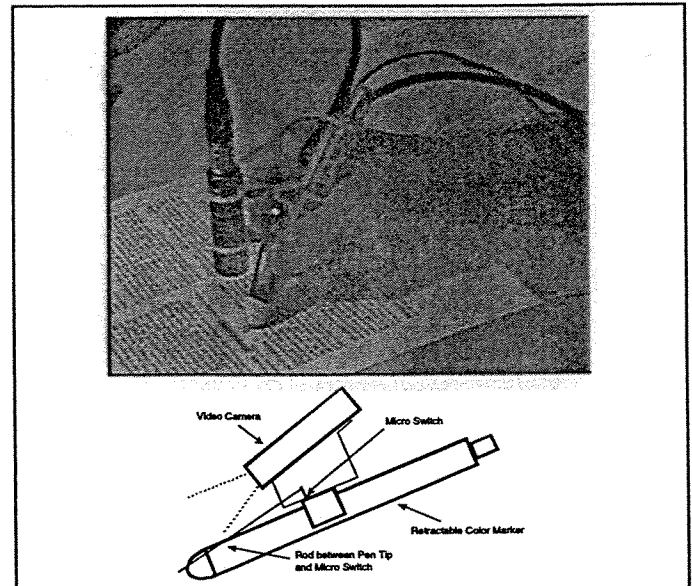


Figure 1. Photo and Schematic of an Early Prototype for an Augmented Highlighter Pen



Figure 2. Second Prototype of Augmented Highlighter Pen (Under Developed by Hitachi Research)

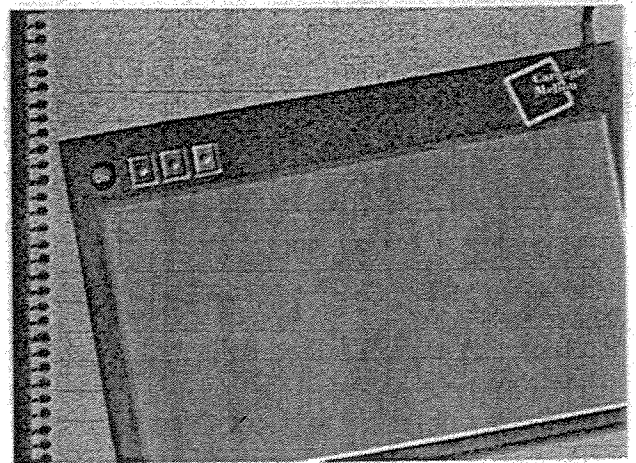


Figure 3. Mockup of a Transparent Overlay Device

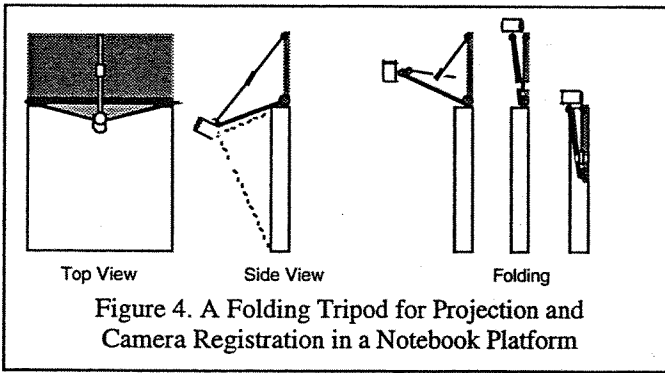


Figure 4. A Folding Tripod for Projection and Camera Registration in a Notebook Platform

union of the paper and LCD surfaces as a unit, much as they would act on the paper surface alone. For example, to select a piece of text under the display, the user might circle the text on the pen sensitive upper surface of the device. In order to provide registration of the LCD image with the paper, the overlay device will be used in conjunction with an input tablet device placed under the paper in a notebook style holder (such as the commercially available CrossPad device [4]). The sensory apparatus from two tablet pens or pucks will be placed at the corners of the device in order to provide accurate position information.

Our third device concept (illustrated in Figure 4) was motivated by ellner's early work on the DigitalDesk system [5], and represents another significantly different point that might be explored in the design space. It employs a camera and projection system for its input and output which is precisely registered with the paper, but can be folded into a notebook. This device is designed to explore concepts for interacting directly on paper in a book-like package. This device will use a projective display to "paint" light on top of paper, combined with drawing tablet input from an ink pen, as well as input from either low- or high-resolution cameras.

As a final illustration of a potential point in the device design space, Figure 5 illustrates another camera-based approach. This device is designed to allow content printed on paper to be "picked up" by the user by pressing the device over the area of interest on paper. It would be constructed in the form factor of a jeweler's magnifying glass (or *loupe*) with contact switches mounted at the corners to signify when the camera is to capture content from the paper. This device might be useful for capturing small pieces of arbitrary content, but shows the most potential for use with controlled content where specialized markers with pre-defined meaning (perhaps even encoding URLs) have been embedded in the paper content and can be "picked up" or activated by the user.

The device concepts shown above illustrate a number of different ways to augment paper with interactive capabilities. Once a range of such device has been constructed, the remaining challenge will be to create interaction techniques that preserve the flexibility and naturalness of the paper medium, while still allowing a

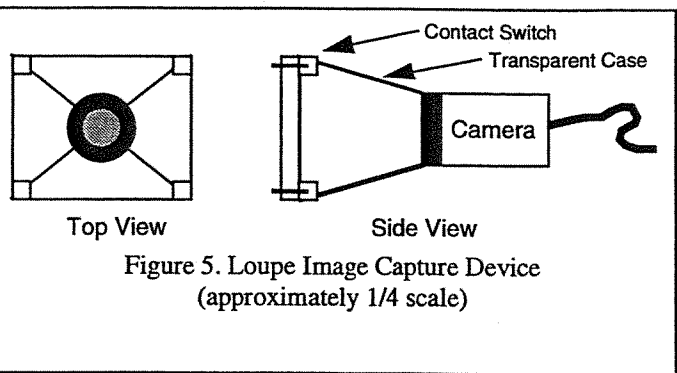


Figure 5. Loupe Image Capture Device (approximately 1/4 scale)

rich set of computational capabilities to be applied. Several possible techniques of this sort have already been illustrated. These include: dynamic "user constructed" interfaces, the use of "printed controls" such as buttons, sliders, and menus, as well as techniques such as "picking up" content or special pre-printed marks and recognizing previous user strokes.

As another example of a potential interaction technique for smart paper interfaces, Figure 6 illustrates a pre-printed form that might be used for a note taking, or meeting minutes application. This form has a series of blank lines for writing in categories of information. For example, the user may wish to tag hand written notes with a person's name, or may want to mark text as an action item, etc. By filling in the form, the user may create whatever categories suit their needs and may create new categories "on the fly". In addition, the form allows groups of categories to be marked as mutually exclusive (by filling in the gap at the left, as has been done to the group of names).

Interaction with this form consists of first establishing a category by writing in a label. Once this is done, the system would assign a color or pattern code to the category (here we assume a projective or overlay display which can add to paper content). To mark a particular section of hand written notes with one or more categories, the user would then "dip" their pen in the colored box at the left of the category, then "brush over"

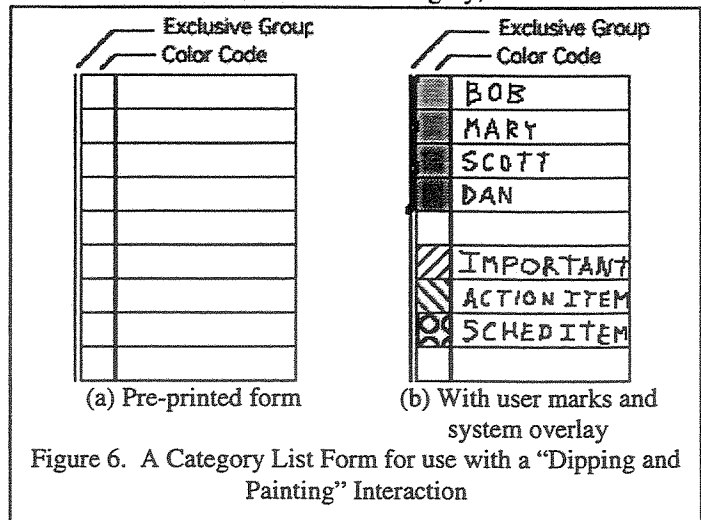


Figure 6. A Category List Form for use with a "Dipping and Painting" Interaction

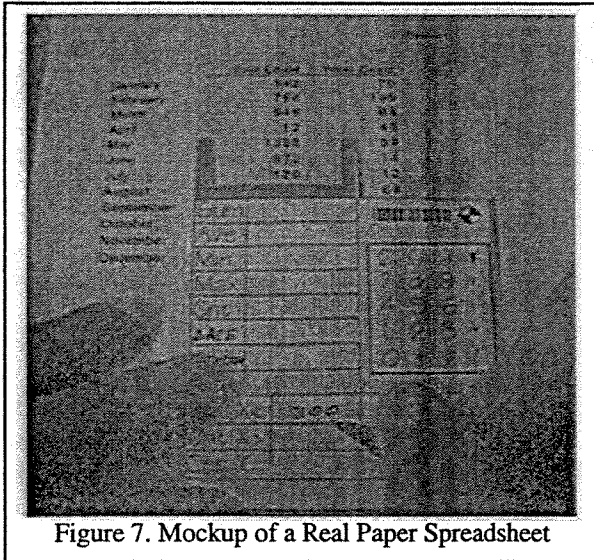


Figure 7. Mockup of a Real Paper Spreadsheet

the text to classify it. Later the user could employ the same form to indicate that selected categories should be highlighted by the system (again using a projective or overlay display).

Figure 7 shows a mockup of another potential example application, a real paper spreadsheet. This application uses a preprinted paper form containing a mix of system provided and user customized interactions. Alignment marks and a bar code have been pre-printed on the form to identify its electronic contents and allow easy tracking of its position. An easily recognizable colored area at the top has been provided to allow the user to indicate a column of numbers from an existing arbitrary content paper document. These numbers will serve as input to the spreadsheet. In addition, affordance for the user such as a paper keyboard and a series of cells with commonly used formulas are provided. Finally, the user can enter new data and additional computations within blank cells. As the user moves the paper spreadsheet over a document, the system would automatically extract columns of data and dynamically present the results of the requested computations using video projected over the

paper.

These techniques illustrate the overall advantage of interacting with smart paper. Because these interactions are mostly free form and user driven, involving simple writing and pointing actions natural to the paper media, it should be nearly as natural as completely free form note taking or paper calculations. With only a small amount of structured help from the system, however, a very useful computational capability can be added which goes significantly beyond the limitations of normal paper.

Finally, these techniques also point to some of the technical challenges facing development of this new interaction modality. For example, to make these examples work properly, the system needs to be able to sense marks made on, and pen movements over the paper, and to add to the display present on the paper. Further, these displays must be properly aligned with the paper contents. In general, a number of such technical challenges will need to be overcome to make this technology practical. However, doing so will open a new interaction modality with significant promise for very natural and flexible interaction.

REFERENCES

1. dpiX Inc., *dpiX Home Page*, <http://www.dpix.com>
2. Arai, T., Aust, D., Hudson, S., "PaperLink: A Technique for Hyperlinking from Real Paper to Electronic Content", *Proceedings of the 1997 SIGCHI Conference*, March 1997.
3. Stone, M.C., Fishkin, K., Bier, E., "The Movable Filter as a User Interface Tool", *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, 1994, pp. 306-312.
4. A. T. Cross Co., *CrossPad Product Page*, <http://www.cross-pcg.com/products/crosspad/pad.html>
5. Wellner, P., "Interacting with Paper on the DigitalDesk", *CACM*, Vol. 36, No. 7, July 1993, pp. 87-96.

Ad-Hoc Networks and Distributed Sensing for Smart Spaces

Ronald A. Iltis, Forrest Brewer, Manos Varvarigos, John J. Shynk, Hua Lee and Daniel Blumenthal

Department of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106

ABSTRACT

Smart spaces will require flexible ad-hoc networking and distributed sensing capability. We describe the physical layer of an ad-hoc network using quasi-synchronous, direct sequence, code division multiple access (QS-CDMA), as well as the wireless network interface to a fiber backbone. In order to minimize power control requirements, a linear decorrelator is proposed which completely rejects multiple user interference with arbitrary time-of-arrival within the QS uncertainty interval. The use of smart antennas for increasing ad-hoc network capacity is discussed. The architecture of a digital VLSI implementation of the decorrelator is also provided. Finally, we discuss the use of backpropagation algorithms for imaging and surveillance applications in the smart space.

1. OVERVIEW – SMART SPACE NETWORKING AND SENSING REQUIREMENTS

A smart space is assumed to consist of a heterogenous network of imaging and acoustic sensors interconnected via wireless modems. The entire smart space itself may be mobile. For example, a search-and-rescue team may form an ad-hoc network as it roams through buildings or in the field. At times, the resulting network may be in contact with a fixed wireless node linked to a high-speed fiber optic network, during which period a large amount of stored information (e.g. compressed video, speech) will be rapidly uploaded. Clearly, the communications requirements for such a network cannot be met by existing commercial systems, which are largely based on a fixed cellular topology. In this paper, we discuss an ad-hoc networking strategy using QS-CDMA, which retains the advantages of direct-sequence spread-spectrum without the need for strict power control.

2. QS-CDMA FOR AD-HOC NETWORKS

2.1. Network Overview

Conventional CDMA systems rely on strict power control to minimize the near-far effect. However, power control cannot be maintained in an ad-hoc network lacking a cellular structure with fixed base stations, and power control errors can have a severe effect on system performance [1],[2]. We have previously proposed a quasi-synchronous CDMA network using local GPS receivers [3],[4] to provide timing. In [3], each user transmits with symbol epochs synchronized to GPS time, and hence reception is quasi-synchronous on the reverse link (mobiles to base). In the ad-hoc QS-CDMA network in [4], GPS position information is used to compensate for propagation delays, so that reception is still quasi-synchronous.

Recently, miniature GPS receivers have been developed that would allow each node in a smart space to have access to accurate time and

position measurements. For example, [5] describes a two-chip GPS receiver that is suggested for implementation in cellular phones, for example. In a smart space formed by a search-and-rescue team, for example, each member could have a CDMA wireless modem and miniature GPS receiver integrated into a helmet. The quasi-synchronous packet radio network (QSPNET) protocol developed in [4] could thus be used in the smart space to implement the physical/transport layer of the ad-hoc network.

The complete QSPNET protocol is described in [4]. In this paper, we focus on the modem architecture and a possible VLSI implementation, rather than on the protocol details.

2.2. QS-CDMA Signal and Decorrelator

Consider a node in the smart space that is receiving data from multiple transmitters. Since each node transmits in synchrony with GPS time, the received waveform is given by

$$r(t) = \sum_{n=1}^N a_n(m)s_n(t - T_n - mT) + n(t). \quad (1)$$

The waveforms $s_n(t)$ are the user spreading codes of common symbol duration T sec. with $L = T/T_c$ binary chips per symbol. The delays satisfy $T_n \in [-MT_s, MT_s]$, where $MT_s \ll T$ due to the QS assumption, and $T_s = T_c/N_c$ is a sampling interval. Note that intersymbol interference (ISI) can be eliminated by inserting a time-guard band of duration $2MT_s$ between transmitted symbols, as suggested in [6]. The terms $a_n(m) \in \mathcal{C}$ represent the data symbols and complex amplitudes corresponding to user n .

A vector model for the received signal, over the m -th symbol duration is obtained by integrating over T_s sec. intervals and sampling. The resulting signal model is similar to that in [3]:

$$r(m) = \sum_{n=1}^N a_n(m)s_n(T_n) + n(m), \quad (2)$$

where $r(m) \in \mathcal{C}^{N_c L}$. It is readily shown that $s_n(T_n) = S_n b_n$, where $b_n \in \mathcal{R}^{2M+1}$ and $S_n \in \mathcal{R}^{N_c L \times (2M+1)}$ is a matrix of code vectors, with delays spanning the QS uncertainty interval. Since any signal vector $s_n(T_n)$ lies in the subspace spanned by the matrix S_n , a decorrelator can be readily constructed [3],[6] that completely rejects multiuser interference that satisfies the QS assumption. To define the decorrelator for detecting user 1, let $S'_1 \in \mathcal{R}^{L N_c \times (N-1)(2M+1)}$ represent the signal matrix

$$S'_1 = [S_2, S_3, \dots, S_N]. \quad (3)$$

Let the projection matrix be defined by

$$P'_1 = S'_1 [(S'_1)^T S'_1]^{-1} (S'_1)^T. \quad (4)$$

The QS decorrelator in [3] is then given by

$$\mathbf{h}_{QS}(T_1) = [\mathbf{I} - \mathbf{P}'_1] \mathbf{s}_1(T_1). \quad (5)$$

The decision variable for differential detection is then given by $x_1(m) = \mathbf{h}_{QS}^H(T_1) \mathbf{r}(m)$. Since \mathbf{P}'_1 is a projection matrix, note that for any undesired user, $\mathbf{P}'_1 \mathbf{S}_n \mathbf{b}_n = \mathbf{S}_n \mathbf{b}_n$. Then since $\mathbf{s}_n(T_n) = \mathbf{S}_n \mathbf{b}_n$, the undesired users are completely rejected, with $\mathbf{h}_{QS}^H \mathbf{s}_n(T_n) = 0$.

2.3. Decorrelator Implementation

A block diagram of a receiver in the ad-hoc network is shown in Figure 1. In order to simultaneously demodulate and estimate the time-of-arrival within the QS uncertainty interval, the receiver computes the following successive correlator outputs during the m -th symbol interval

$$y(l) = \frac{1}{\|\mathbf{h}_{QS}(lT_s)\|} \mathbf{h}_{QS}^H(lT_s) \mathbf{r}(m), \quad (6)$$

for $l = -M, -M+1, M$. The correct time-of-arrival lT_s is chosen as $\hat{l} = \arg \max_l |y(l)|^2$. This strategy can be shown to approximate the maximum-likelihood estimator for the delay T_1 [3].

The primary advantage of the decorrelator receiver in Figure 1 is that the vector \mathbf{h}_{QS} is solely a function of the assigned codes in the network. In the QSPNET protocol [4], the control channel is used to assign codes, and hence each receiver knows which users it is to demodulate, and which users need to be rejected. The corresponding decorrelator \mathbf{h}_{QS} can thus be obtained from a look-up table, and only needs to be changed when users enter or leave the ad-hoc network.

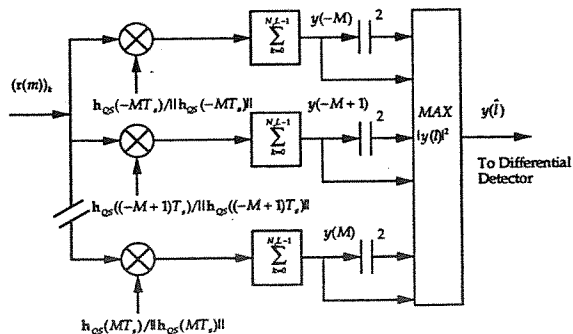


Figure 1: Decorrelator for the ad-hoc network with synchronization.

3. OVERVIEW OF THE QSPNET PROTOCOL

The QSPNET protocol suitable for an ad-hoc network in a smart space is now reviewed. The channel will be divided into two parts: the *data channel* used for data transmission, and the *control channel* used for making reservations and providing other control information. The time axis is divided into frames, each containing a fixed number of packet time slots, which are known to the users through the GPS clock.

The data and the control channel are implemented by using two different types of codes, to be referred to as d-codes and c-codes. The c-

code reserves time-slots/codes and provides synchronization, feedback and GPS location information. The actual user data is modulated onto a direct-sequence (DS) d-code for transmission. When a mobile successfully transmits a c-code packet, it can reserve the corresponding (slot, d-code) pair of the next frame. This can be done either by using a field in the packet header saying that another packet is to follow, or more simply and less efficiently, by automatically reserving that (slot, d-code) pair in each subsequent frame until it is first empty. Immediately upon the successful receipt of a c-code reservation request, the intended receiver transmits an ACK using the same c-code. ACK transmissions are also used to acknowledge subsequent data packets, in order to minimize hidden terminal problems in the network. After the end of the reservation period, the given (slot, d-code) is open for contention.

By using a simple correlator detector, a given user can listen to the ACK packets on the control channel to determine if a code is in use or is open to contention. If two users transmit simultaneously using the same c-code, a collision occurs and they have to retransmit after a random interval, or as required by the priority and congestion control protocols.

In order to simplify the reservation overhead, avoid the situation where some heavy users monopolize the available capacity, and at the same time provide transmission rate guarantees to applications that require them, we will use a variation of the above protocol, where each node "owns" a particular channel (or set of channels). When an owner is not using its channel, other nodes can capture it by contention, but when the owner wants it back, it simply transmits a request on that (slot, c-code) pair, and all sources hearing an ACK or collision in the subsequent reverse minislot are forbidden to use that channel on the next frame, letting the owner capture it. In this way, each node is guaranteed a minimum transmission rate even in the presence of congestion. The problem of deleting and adding new owners, which arises from the mobility of the users, can be addressed by having a node periodically affirm its ownership of a channel through the transmission of (dummy) request packets, possibly forcing a collision. When two mobiles that own the same channel approach each other, one of them will have to abandon the channel (this is similar to the handover problem in cellular networks.) A user who loses, due to this slow drifting of the mobiles, a channel that it owns, can capture a new one by transmitting on a (slot, c-code/d-code) channel that is not owned by other nodes, forcing any temporary user to cease transmission.

4. SMART ANTENNA IMPLEMENTATION

In order to augment the capacity of QSPNET, we propose incorporating smart antennas that beamform in the direction of the desired transmitter and null the other users that contribute to the cochannel interference. Smart antennas consist of an array of antenna elements that introduce a directional capability that is not possible with a single antenna element. The directional gain is controlled by a set of adaptive coefficients that can be adjusted in real time using the statistics of the received signals [7], [8]. The array is self-adjusting and can track source variations in an automatic manner. Blind adaptation of the receiver can be implemented by using known invariant properties of the transmitted signals, without specific knowledge of the data. Alternatively, a known data sequence may be used to initially compute the coefficients during start-up, which can then be adjusted to track slow variations in the signal parameters.

One architecture being considered is based on the principle of successive interference cancellation (SIC) [9], [10], where a multistage system sequentially recovers the cochannel users from the received signals. Each stage provides an estimate of one source by using the decorrelator previously discussed. This estimate is used to regenerate the portion of the received signal due to that user. The detected data bits then modulate a carrier with the appropriate delay, amplitude, and phase. The remodulated signal is subtracted from the original received signals (i.e., interference cancellation), and the result becomes the input of the second stage. This procedure is repeated for each stage in the multistage architecture until all the cochannel users have been demodulated. This approach is reminiscent of the multistage constant modulus (CM) array that was developed to separate cochannel analog FM signals (see, e.g., [7]).

In the SIC, the first stage estimates and removes the strongest user signal (at the receiver) because this also removes the most amount of interference from the other users. Moreover, the strongest signal is also the easiest one to remove (as it was in the CM array). Obviously, the first signal does not benefit from this cancellation approach, but since it is the strongest signal, it will be the one most accurately detected up front. The weakest signal benefits most from this approach. To achieve the best performance, the received signals should be ordered and detected in decreasing order of power.

Exact parameter estimation is required to regenerate the received user signals; otherwise, we can expect some noise amplification rather than the desired signal cancellation. Also, because of the cascade structure, bit delays will be incurred for every user cancelled, thereby imposing a practical limit on the number of users that may be cancelled. Imperfect cancellation also imposes a limit on the number of cancelled users. Thus, we might limit the number of stages to remove only the strongest users, and employ a conventional detector at the final output to extract the desired signal. By that stage, the remaining users should have relatively equal power levels.

5. FIBER BACKBONES FOR DISTRIBUTED WIRELESS NETWORKS

As the size of wireless cells decreases (picocells) and the bandwidth of traffic increases, a bottleneck exists when transmitting data across multiple wireless cells. In addition to the bandwidth bottleneck, access to limited bandwidth can lead to intolerable latencies for communications across multiple cells. In applications where response time for collection of information is critical, network bandwidth and latency must both be optimized.

Fiber optic networks can efficiently address the bandwidth and latency problem in distributed wireless networks and applications. The fiber network can act as a backbone to interconnect micro and pico cells [11], thereby reducing the need for expensive base stations at every cell [12] and reducing bottlenecks for communications across multiple cells.

Interconnecting multiple cells via a fiber backbone can take advantage of high bandwidth optical multiplexing techniques like wavelength division multiplexing (WDM) to guarantee access, reduce latency, and match the optical channel bit rate efficiently to low cost wireless electronics. Additionally, optical subcarrier techniques allow digitally modulated RF signals to be directly impressed onto an optical carrier [13] for transport and extraction at another cell. Optical subcarrier techniques also allow digital and analog transmission

to be combined with multiplexing at both the optical and RF level.

It is our intent to investigate the application of fiber-optic network backbones to serve in a smart space environment where data is collected by distributed wireless sensors. The fiber channel will be used to move data across multiple cells and to primary data collection points where data fusion and other functions are performed.

6. VLSI IMPLEMENTATION OF THE QS-CDMA DECORRELATOR

6.1. Decorrelator Architecture

Most components of a direct-sequence spread-spectrum modem are available off-the-shelf, including RF amplifiers, downconverters, and IF filters. However, commercially available spread-spectrum correlators are restricted to less than 8 bit wide inputs, and at most, three level quantized correlating waveforms. These correlators are well-suited to single user or IS-95 applications, where the correlating waveforms are the binary user sequences themselves. In contrast, the decorrelator in (6) corresponds to a non-binary correlating waveform. Hence, we concentrate on a VLSI implementation of the decorrelator in VLSI.

The quasi-synchronous nature of the proposed network and the need to accommodate large changes in signal level place special demand on the downconverter and decorrelator designs. In particular, the received sample and decorrelator coefficient resolution must both be increased to allow sufficient cancellation of unwanted noise and signal sources. For the purposes of this paper, we shall assume 8-bit resolution for both the down converted chip samples and the decorrelator coefficients. This assumption can easily be changed to accommodate differing use requirements with virtually no change in receiver architecture. In a conventional approach, this resolution would lead to a very large design and far too much power dissipation for a portable transceiver. In the quasi-synchronous system, however, we need only compute correlation sums for $-M, \dots, M$ shifts about the expected symbol boundary. Thus, in $(LN_c + 2M + 1)$ sample periods, we need compute only $2M + 1$ correlations. Since each correlation is the sum of LN_c separate multiplications, and since the coefficients $(h_{QS})_k$ are not simple shifts of each other, we must compute $(2M + 1)LN_c$ multiplies and a similar number of adds. This can be performed by $2M + 1$ parallel multiply/accumulate devices, each operating at the sample rate $f_s = N_c/T_c$. Alternatively, faster multipliers could be used in a time-multiplexed fashion to trade off receiver area for power. Interestingly, the addition of more taps which could be used to increase the processing gain adds only marginally to the area of the design, since the number of computations per sample interval does not change. Adding taps with a fixed constraint of symbol rate (decreasing chip sample period) does lead to increased power dissipation roughly proportional to the change in sample frequency.

A block diagram for the decorrelator circuit is shown below in Figure 2. In this design, the 8-bit chips are sequentially delayed and separately correlated against the $(h_{QS})_k$ coefficients stored in either RAM (if needed for greater flexibility) or cheaper ROM which is possible given the intended protocol for a fixed family of low-power transceivers. Note that all the multiplies and accumulate functions run at the sample rate f_s , but that the square-law selector need only run at the symbol rate $1/T$. Other receiver high-rate units are the IF modulator and filter which comprise about the same power dissipation as 2 tap multiply/accumulate sections. If implemented in an

Each 8x8 multiply:	50 μ W/MHz	75k μ m ²	18nS delay
Each 23-bit add:	16 μ W/MHz	21k μ m ²	18nS delay
Each 8-bit latch:	5 μ W/MHz	10k μ m ²	0.9nS delay

Table 1: Power requirements - Decorrelator.

inexpensive bulk 0.5 μ m 3.3V CMOS process, power estimation is summarized in Table 1.

Using the figure in Table 1, at a 20 MHz sample rate, we have a total of 1.7 mW/tap or about 14 mW for a 7 tap decorrelator. Given 128 samples/symbol we get a 156 ksymbol/sec channel with a processing gain of 18dB. Apart from the local high-resolution clock and IF down converters, the remaining logic may run at a much lower (symbol) rate. Thus, receiver estimated power is about 20 mW for the high speed circuitry, and another 15 – 20 mW interconnection loss. Given the low bandwidth outputs of the receiver, its pad I/O should be under 8 mW for a total chip dissipation in the neighborhood of 50 mW. Lower power dissipation is available in finer technologies – 0.35 or even 0.25 μ m is relatively inexpensive. However, one must remember that the A/D providing the samples is likely to dissipate at least as much power as the proposed design. This could be lowered by integration of the A/D itself in a mixed-signal technology implementation.

6.2. UCSB Capabilities in VLSI/CAD

UCSB has a very active VLSI/CAD design group with capabilities and infrastructure in bulk Silicon CMOS provided by Synopsys, Mentor, Cadence, Duet and other design software suites as well as in-house high-level, logic synthesis, and test applications. Recent Silicon designs have been 40 MByte/Sec Des/Data Compression design in 0.8 μ m CMOS, and a 50 MIPS PIC instruction microcontroller in 0.5 μ m CMOS. Other recent designs include 400 Mhz current mode I/O test chips and 100Mhz integrated DRAM parametric test for MCM applications. A particular strength of the department is in ultra-high speed integration (50GHz+) provided by in-house fabrication abilities for MSI circuits in advanced GaAlInAs and MBE grown wafers. This expertise is integrated with the digital design group in such recent designs as a 40 Gb/Sec fiber channel encoder/decoder set implemented in CML using the Rockwell ATE GaAs process.

7. SENSING SYSTEMS AND IMAGE FORMATION ALGORITHMS

The focus of the development of sensing devices for smart spaces applications has been mainly placed upon traditional optical or infrared camera systems because of the commonality to the human vision systems and similarities in terms of system configurations and image formation procedure. Yet, advanced imaging systems have spanned the domain of applications, in both civil and military sectors, far beyond the traditional optical-infrared range. Especially in recent years, microwave, acoustic, and ultrasound sensing systems have been playing increasingly important roles. In many situations, these systems are uniquely effective while the propagation media are not feasible for visible light or infrared waves. Therefore, it is crucial to incorporate a wide range of sensing systems into the smart spaces technology in order to realize and optimize its potential.

As we expand the sensing modality of the smart spaces technology to include microwave, acoustics, and ultrasound, the image forma-

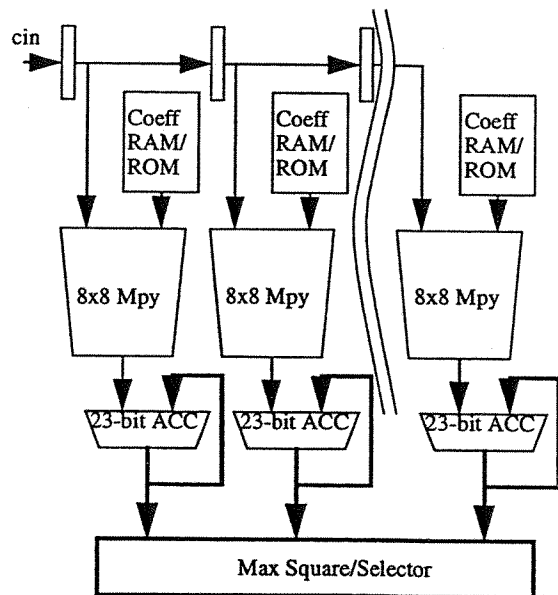


Figure 2. Decorrelator Block Diagram

Figure 2: VLSI Decorrelator Implementation.

tion process becomes complex. To achieve high-resolution imaging, image reconstruction algorithms becomes a subject of critical importance. Among the image formation techniques, backward propagation based algorithms are capable of performing the tasks with the following key features.

- high-resolution resolving capability,
- high degree of stability and sensitivity,
- unified algorithm structures,
- near and far-field imaging,
- monostatic and bistatic operations,
- parallel-processing architecture, and
- consistent algorithm performance.

In addition, backward propagation based algorithms have the unique structure for which object recognition procedure can be directly integrated in the process instead of functioning as a post-processing task. Most importantly, these algorithms can be expanded, because of the sound physical and mathematical models, to function in dynamic operating modes where target or propagating media become time varying.

References

1. R. Cameron and B. Woerner, "Performance analysis of CDMA with imperfect power control," *IEEE Transactions on Communications*, vol. 44, pp. 777–781, July 1996.
2. F. D. Prisco and F. Sestini, "Effects of imperfect power control and user mobility on a CDMA cellular network," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 1809–1817, Dec. 1996.
3. R. Iltis, "Demodulation and code acquisition using decorrelator detectors for QS-CDMA," *IEEE Transactions on Communications*, vol. 44, pp. 1553–1560, Nov. 1996.

4. A. Banerjee, R. Iltis, and E. Varvarigos, "Performance evaluation for a quasi-synchronous packet radio network (QSP-NET)." Submitted to the *IEEE/ACM Transactions on Networking*, 1997.
5. G. Turetzky, "Bringing GPS chipsets to mainstream products," in *WESCON/97 Conference Proceedings*, (Santa Clara, CA), pp. 8–11, Nov. 1997.
6. F. van Heeswyk, D. Falconer, and A. Sheikh, "A delay independent decorrelating detector for quasi-synchronous CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 1619–26, Oct. 1996.
7. J. J. Shynk and R. P. Gooch, "The constant modulus array for cochannel signal copy and direction finding," *IEEE Transactions on Signal Processing*, vol. 44, pp. 652–660, Mar. 1996.
8. A. V. Keerthi and J. J. Shynk, "Separation of cochannel signals in TDMA mobile radio," *IEEE Transactions on Signal Processing*, vol. 46, Oct. 1998.
9. A. J. Viterbi, "Very low rate convolutional codes for maximum theoretical performance of spread-spectrum multiple-access channels," *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 641–649, May 1990.
10. S. Moshavi, "Multi-user detection for DS-SS-CDMA communications," *IEEE Communications Magazine*, vol. 34, pp. 124–136, Oct. 1996.
11. W. Way, "Optical fiber-based microcellular systems: An overview," *IEICE Trans. Commun.*, vol. E76-B, pp. 1091–1093, Sept. 1993.
12. O. Tonguz and H. Jung, "Personal communications access networks using subcarrier multiplexed optical links," *IEEE Journal of Lightwave Technology*, vol. 14, pp. 1400–1409, June 1996.
13. D. Blumenthal, J. Laskar, R. Gaudino, and H. Sangwoo, "Fiber-optic links supporting baseband data and subcarrier multiplexed control channels and the impact of mmic photonic/microwave interfaces," *IEEE Trans. on Micro. Theory and Techn.*, vol. 45, pp. 1443–1452, Aug. 1997.

Smart Information Spaces: Managing Personal and Collaborative Histories

Francis Kubala, Sean Colbath, John Makhoul
{fkubala,scolbath,makhoul}@bbn.com

BBN Technologies, GTE Internetworking
70 Fawcett Street, Cambridge MA, 02138

Abstract

We assume a future in which access to information is ubiquitous. *Smart spaces* will extend the reach of networked information stores and services to people as they move freely about in the course of their occupational or personal lives. Groups of mobile people will use the smart space fabric to stay connected to each other while they collaborate on distributed tasks. In order for this assumed future to become a reality, it will be necessary to have comprehensive tools in place to manage the voluminous and diverse information flow that will result from ubiquitous access. Each person will experience a rapid inflow of information that will accumulate too quickly to be managed by hand. In this paper, we anticipate problems in information management that will be encountered in a world of smart spaces and mobile people and introduce a vision for *Smart Information Spaces* that are designed to shoulder the burden of managing rapidly accumulating personal and collaborative histories.

1. Introduction

Private information spaces (digital stores or archives belonging to individuals or collaborative groups) are growing quickly as computer assisted transactions become more common in everyday life. At the same time, digital mass storage densities are increasing faster than the rate predicted by Moore's Law for transistor densities in microprocessors. There is every reason to believe that these two trends will continue well into the future. We assume that, at some point in the near future, storage capacity will be essentially infinite and of negligible cost so that anything that might be useful later can be saved. At that point, the dominant issues will be organization and retrieval of the archived data.

In the next section, we examine data management issues for four sources of data that are likely to be part of personal history archives of the future: email, voicemail, Internet retrievals, and videoconferences. Email and Internet retrievals are primarily text based whereas audio recordings of speech constitute the bulk of voicemail and videoconference data. The inadequacy of today's tools to deal with data from these four sources provides the motivation for a vision of a *Smart Information Space* that is presented in section 3. In section 4, we briefly review some of the major technical obstacles that must be overcome to achieve this vision and in the last section, we mention several research efforts that are beginning to develop capabilities needed to overcome these obstacles.

2. Problems Managing Personal Histories Today

Email clients generally support only simple archives of outgoing and incoming messages organized around external attributes of the messages such as sender, subject, and date. Management of an email archive by message content is done by hand, if at all. In addition, email archives are maintained in client-dependent formats that segregate the archives from the rest of a person's documents and messages so they must be managed separately with their own client-specific tools.

The state of today's voicemail systems is much worse. The only voicemail history management tools available are play and delete! Audio data is simply not usable as a retrievable information source, so it is discarded at the earliest opportunity. In fact, if you are too slow to delete your voicemail

history, the system will do it for you or deny you future storage services until you do. This state of affairs discourages the passing of information among people by voice. If detailed information is conveyed in voicemail, it is assumed that the listener will write down the information and deal with it as text from then on. In addition, voicemail for most people is completely segregated from all other electronic information, which further reduces its value as a source of persistent information

This lack of support for managing personal email and voicemail histories is striking since these technologies have been in widespread use for more than a decade. People who use computers daily in their work are already experiencing difficulty maintaining their email histories with today's primitive tools. This annoying problem will grow to become an overwhelming barrier in the future world of *smart space* technology that we envision in which electronic interactions among people become commonplace and access to information is ubiquitous. We already have more access to information than we can manage.

Email and voicemail management problems are small potatoes compared to the task of organizing one's total flow of information, including interactions over the Internet and in video-conferences. The easy availability of the vast content of the World Wide Web has created a new source of documents and transactions streaming into private archives with little support for constructing, maintaining, and accessing them. Browsers typically do provide a 'bookmarking' facility to easily create pointers to locations on remote Internet servers that the user may want to visit again. But as the bookmarks accumulate, they present their own management problems. Furthermore, bookmarks become orphaned if the location or name of the documents change.

Data fetched via the Internet are typically cached in temporary hidden directories on the local client machine to facilitate display and navigation of the data. The data caches are not designed to persist, however. In fact, they are routinely swept clean when storage space runs out, as if the contents of the cache were of fleeting interest only. This reinforces the notion that information contained in retrievals from the Internet is for the most part

ephemeral. Analogous to voicemail, if end-users want to preserve data for later use, then they need to perform some manual step such as saving the display to a file somewhere in their personal data storage space.

This notion of web retrievals as ephemeral is ironic since the cache contained every document that someone thought important enough to extract from the public web and then examine. These items are very likely to be sought again at a later time or by another person in a collaborative role.

Within collaborative groups, many documents are created and passed around electronically. Sometimes, intranets are created to try to put this important material within the reach of a group, but these archives require a good deal of labor to create and maintain and, if they become very large, they are difficult to use.

Another potentially huge source of information in the near future is video-conferencing. Today's typical desktop PCs are powerful enough to handle conferencing software such as Microsoft's NetMeeting, which is already widely distributed due to its bundling with Windows OS. The rate-limiting stage now is network bandwidth. As soon as high capacity digital services become common, the volume of information ready to accumulate in private information spaces may increase dramatically. As people become more familiar and comfortable passing multi-media information around and conducting business and live conferences over networks, the problem of managing private information spaces will become of paramount importance. Most of the information in videoconferences is contained in the audio track, but today's tools for recovering information from audio are just beginning to be developed.

We have considered only four sources of information here: email, voicemail, Internet retrievals, and videoconferences. Two are text based; two are dominated by audio information. It goes without saying that there are many other kinds of information that will collect in electronic stores of the future such as purchase and payment records, medical records, contractual and other legal documents, etc. It's also obvious that the content of private information spaces is of great value to

individuals and collaborative groups. They form complete records of past activities and encompass the entire electronic memory of an individual or group. Great effort is invested today in organizing and maintaining private information spaces by hand, but the resulting archives are typically only small static snapshots of the entire relevant information space. Without a self-organizing data management system, most of the information contained in the daily interactions among people will be inaccessible to anyone not present at the moment of its creation. These vast stores of ephemeral data will be as underutilized as voicemail is today.

3. *Envisioned Solution*

We envision a Smart Information Space (SIS) that keeps our entire individual or collaborative histories visible, in context, and accessible at all times. It should never discard anything by default. The SIS will be self-organizing so that no end-user input is required to save data as it is produced, captured, or redirected. A SIS will free us from the tyranny of folder creation and file naming. Folders and filenames will always be optional. Retrieval tools will be powerful and simple enough to convince the user to rely on the system-generated organization most of the time.

Any document, recording, or transaction conducted within the scope of a SIS will be captured by it and transparently linked to the archives. This linkage to the archive will be made on the basis of the content of the source and any other external features of the source that can be captured automatically. Content-based features will include topics, proper names (of people, organizations, locations), and speaker identities. External features will include time of capture and source of the data as well as a transaction record indicating how the data was acquired and how it was used. For example, a document obtained as the result of a query would retain that query as a descriptive feature. If the document was examined, or printed, or forwarded, a record of these interactions would be kept. Such transaction records can help the system determine which data are most important.

These content-based and external features will be used for indexing and organizing the data. Data sharing features in common can be viewed as a group. The same features will permit flexible and selective information retrieval and extraction from the archive due to their specificity and redundancy.

In the absence of explicit directives from users, construction and maintenance of the archive will be completely automatic. Capture and linkage of new information to the archive will be of no concern to the user unless one deliberately chooses to control it. Although the SIS is designed to be self-organizing, it will never block an end-user from adding or changing annotations explicitly. The SIS will make it easy to capture the precise knowledge of the user whenever possible. Users will be able to explicitly control links among items in the archive and add their own features and annotations to them if they desire. This will permit users to flag sensitive or time-critical data and ensure that it is immediately visible to all members of the SIS. These local manual modifications will be propagated throughout the SIS under the control of the user.

The SIS will handle multiple modalities transparently. All email, voicemail, original documents, web-searches, and conferences will be linked together and made visible under a common set of navigational and retrieval facilities. A SIS will dynamically construct and maintain 'conversational' threads, even across modalities. Users of the SIS will be able to recover the state of a previous conversation or interaction thread and continue it or view its evolution over time

Threads may not be simple linear sequences of temporally distinct events. At some points, they may thicken into dense webs of information that were brought to the SIS at the same time. For some items in a thread, temporal order may not be the most important feature so they will join the thread at many points where they share features with other data. Distinct threads will intersect around common topics. The SIS will provide visualization graphics to facilitate efficient navigation and retrieval of items embedded in the thread fabric.

The SIS will organize duplicate and versioned data in a manner analogous to threads. The most useful

version is usually the latest one, so it will be used by default as the active placeholder for the group of related documents.

Written or spoken language will be treated uniformly whenever possible. The end-user will be able to locate information in either modality by specifying the topic of interest, for instance. At the same time, the unique features of both modes will be exploited. Speech, for example, contains speaker information that can be used as a content-based feature of the words spoken. Written language contains case and punctuation information that has no direct analog in speech.

The SIS will also support a change in modality when one is strongly preferred over the one that is available. For instance, addresses and telephone numbers are much more useful as text in which form they can be embedded in forms and launched in applications. To facilitate moving spoken entities to text, the SIS will have special purpose audio-to-text transducers for these exceptionally useful cases. The end-user will be able to drag an audio sample containing a phone number into one of these transducers and have a text string produced that can be dropped into an address book application.

The SIS will manage the significant storage issues arising from the huge sizes of private information spaces in which little is discarded. Contents of a SIS will be stored in a manner appropriate to their importance, sensitivity, and timeliness. If selective removal of data becomes necessary, then the system will help the user select the data to prune by identifying the oldest, least important, or most redundant data in the archives.

The SIS will manage data-staging requirements imposed by smart space devices of widely differing capabilities. Readily accessible high-speed caches will contain only the most important and recently accessed parts of an archive. Less important material will be backgrounded to less accessible (less expensive, more distant) media. Critical but inactive material will be migrated to safe long-term storage. These hierarchical storage management functions will operate well below the view of the SIS members by default

The SIS will allow groups of people to construct and manage shared histories as easily as single-user repositories. Personal and group histories may overlap and the SIS will need to manage access control mechanisms among its members. The shared SIS will become the easiest way to construct and manage Intranets that contain rapidly changing information.

With Smart Information Space technology in place, users will be free to concentrate entirely on the communication act without concern for how the interaction will be captured and saved. They will have complete records of all of their electronic interactions since nothing will be discarded. And they will enjoy efficient reliable access to their private histories through thread visualization tools and hierarchical storage of the archive that place the most important items in the highest capacity channel.

Smart Information Spaces automate most of the manual work required to maintain records of today's networked communications. They extend the capability of today's primitive archiving tools to permit completely hands-free construction and maintenance of exhaustive archives that capture all communications between people. They provide the reassurance that anything ever sent or received can be retrieved efficiently on demand. Groups of people working in collaboration within a team, business, or institution would benefit enormously by having the organization and maintenance of their communications histories performed largely automatically. A SIS would change the default of publishing only what someone deems worth the trouble of placing on an Intranet, to that of publishing everything

Complete electronic records are of importance to almost everyone, but they are absolutely vital to corporations and government institutions and the military where vast amounts of time-critical and sensitive information are generated and handled daily. In many large collaborative undertakings, such as asset management or logistics planning, efficiency is gated by the rate at which information can be distributed and digested by members of the enterprise. Any improvement in the efficiency of information sharing has a direct effect on the efficiency of the entire operation. Smart

Information Spaces are designed to materially improve the means of disseminating and using large volumes of information from diverse sources.

4. Major Technical Obstacles

Automated construction and maintenance of an archive will require sophisticated means of characterizing documents and recordings so that they can be linked to an archive in ways that make sense visually and permit effective retrieval. Navigation within a SIS and retrieval of a particular item will need to be efficient and very reliable. The notion of saving everything is compelling only if one can find relevant information quickly when it is needed. If an item cannot be found after a reasonable amount of search effort, there is little point in saving it in the first place. Moreover, if retrieval is not very easy and reliable, users will not trust the system to organize the data automatically and will return to self-defeating manual organization strategies.

A Smart Information Space needs to form a notion of relevance, similarity, importance, criticality, timeliness, and sensitivity for each item in the archive in order to present large amounts of information in a visually useful fashion and to manage the hierarchical storage of the archive. Each assessment made by the SIS should have a confidence measure associated with it to assist in visualizing the archive along the most accurate dimensions. Much of this capability exists only in rudimentary form today.

New capabilities will need to be developed to support construction of discussion threads that may be multi-modal. These capabilities will need to work on spoken speech as well as they do on text. They should also operate in the same manner on both. Today's capabilities on speech are considerably less accurate than on text. Construction of general 'discussion' threads will require abilities to distinguish novel threads from existing ones.

The SIS will need to be able to detect and track new topics. The notion of novel topics (and threads) is not well defined. It will also be necessary to find a means of inferring new topics from minimally

annotated training data. Useful measures of similarity between multi-modal documents and recordings will need to be found to create links among them when they share common features.

SIS history archives will need to be shared. Operations on shared archives should be identical to those on private archives for the same functionality. A single SIS may need to manage several distinct but overlapping archives that differ in sensitivity and membership. Links to such families of archives may need to be fail-safe with respect to exposure of restricted material. Privacy and access control issues may be some of the most difficult problems that the SIS will need to solve. Users will come to rely on the SIS to maintain the archives only to the degree that their experience proves this a successful strategy. Failure recovery strategies will need to be biased toward risk avoidance.

5. Preliminary Research

One approach to self-organizing data archives, called *Lifestreams*, is currently under development [2,3]. This work has grown from the vision of David Gelernter at Yale University. It embodies several of the most important ideas presented here. Under the Lifestreams model, all data is saved in a global time-indexed stream that can be effectively viewed and navigated by content-based and external features of the data. The model is viewed as a human-computer interaction metaphor that can completely replace the desktop metaphor in common use today. In the future, the Lifestreams model will support audio data as well.

Work in advanced visualization capabilities for large personal history archives is underway at the University of Maryland [8]. This work is concerned with easing the handling and interpretation of highly structured personal data such as medical records. Dynamic views of the data are activated by adjustable sliders that emphasize different dimensions of the data under the user's control.

Current work is underway at BBN Technologies to index meeting and news audio data under a project called *Rough'n'Ready*. We are building a meeting

recorder and browser that automatically produces a ROUGH transcription of what was said, along with a content-based structural summarization of the audio recording, which is READY for browsing. The summarization meta-data provides a framework for data visualization and an index for efficient navigation of large audio archives. The basis of the structural summarization is the automatic transcription produced by our state-of-the-art large vocabulary speech recognition system (BYBLOS) [6].

On top of the transcription, we have added annotations from three component technologies developed at BBN – speaker identification [4], named entity extraction [1,7], and topic classification [5,9]. Our focus is on audio data but the approaches we use for named entity extraction and topic classification are applicable to text sources as well.

The Rough'n'Ready browser currently allows the user to navigate through large audio archives and retrieve specific passages by means of a multi-valued query composed of any combination of topic, named entity, and speaker identity. The ability to specify multiple features in combination makes the query more selective than a simple full text search. The current implementation of the browser would provide a logical interface for the kinds of advanced SIS capabilities on audio data that are envisioned here.

Summary

As individuals and organizations, we produce much more data than we can manage with today's tools. If we wish to keep a history of our transactions, we are confronted with the difficult problem of organizing it so that the information in it can be recovered when needed. Current technology has progressed to the point where we can begin to envision building a *Smart Information Space* that would collect all documents and transactions among people and maintain them in self-organizing history archives. These personal and collaborative histories would persist over time and be searchable with tools that work on speech as well as text. The tools would operate upon content-based features

and external attributes generated automatically by the system.

References

- [1] D. Bikel, S. Miller, R. Schwartz, R. Weischedel, "NYMBLE: A High-Performance Learning Name-finder," Proceedings of the Fifth Conference on Applied Natural Language Processing, Association for Computational Linguistics, 1997, pp. 194-201.
- [2] E. Freeman, S. Fertig, "Lifestreams: Organizing your Electronic Life," AAAI Fall Symposium: AI Applications in Knowledge Navigation and Retrieval, November, 1995, Cambridge MA.
- [3] E. Freeman, D. Gelernter, "Lifestreams: A Storage Model for Personal Data," ACM SIGMOD Bulletin, March, 1996.
- [4] H. Gish, M. Schmidt, "Text-Independent Speaker Identification," IEEE Signal Processing Magazine, October 1994, pp. 18-32.
- [5] T. Imai, R. Schwartz, F. Kubala, L. Nguyen, "Improved Topic Discrimination of Broadcast News Using a Model of Multiple Simultaneous topics," Proceedings of ICASSP 97, Munich, Germany, April 1997, pp. 727-730.
- [6] F. Kubala, H. Jin, S. Matsoukas, L. Nguyen, R. Schwartz, J. Makhoul, "Advances in Transcription of Broadcast News," Proceedings of Eurospeech 97, Rhodes, Greece, September 1997, pp. 927-930.
- [7] F. Kubala, R. Schwartz, R. Stone, R. Weischedel, "Named Entity Extraction from Speech," Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop, Lansdowne VA, February 1998.
- [8] C. Plaisant, B. Shneiderman, "An Information Architecture to Support the Visualization of Personal Histories," Human Computer Interaction Laboratory Technical Report: UMIACS-TR-97-87, University of Maryland Institute of Advanced Computer Studies, 1997.
- [9] R. Schwartz, T. Imai, F. Kubala, L. Nguyen, J. Makhoul, "A Maximum Likelihood Model for Topic Classification of Broadcast News," Proceedings of Eurospeech 97, Rhodes, Greece, September 1997, pp. 1455-1458.

Important Technology Components in Smart Spaces

Chiman Kwan, Don Myers, Roger Xu, and Len Haynes

Intelligent Automation Incorporated
2 Research Place
Suite 202
Rockville, MD 20850
ckwan@i-a-i.com

Abstract

Here we summarize a few component technologies that are important in smart space applications. They are: 1) active speech enhancement by eliminating background noise for speech recognition; 2) active beam steering technology for sensor array steering; and 3) face recognition using cameras.

1. Speech Enhancement

Two approaches to active speech enhancement are presented in this paper. One uses two microphones and the other uses only one microphone. Experimental results show that the one uses only one microphone performs well.

Approach 1: Two microphone approach that uses the spatial correlation between the signals

The system concept is shown in Figure 1 below. It is based on the assumption that two microphones placed near each other will pick up highly correlated noise, but if one is placed directly in front of a speaker's mouth, than microphone will pick up that user's speech with a higher amplitude than the other microphone, nearby but not directly in front of the speaker's mouth.

In Figure 1, SPLF (speech-like pass filter) is a bandpass filter with a frequency range from 200 Hz to 4000 Hz. This filter passes speech without attenuation, and passes noise in this frequency-band, but eliminates noise at other frequencies.

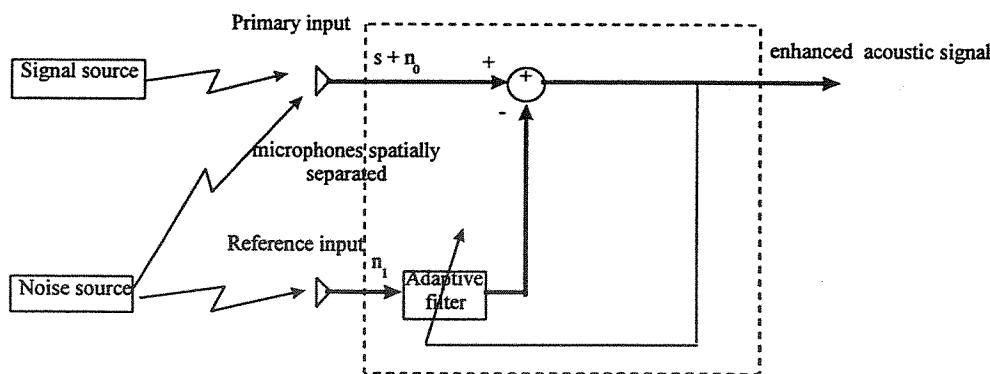


Figure 1 (a) System configuration.

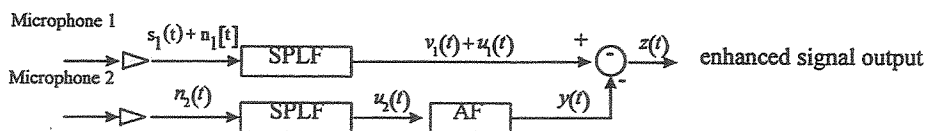


Figure 1 (b): Schematic of the noise rejection system

Experimental Setup

Two computers were used to perform the experiment, one for playing the helicopter noise through 2 loudspeakers at different locations in a small room and the other for recording 2 microphone signals. One microphone is

omni-directional and the other is unidirectional. The unidirectional mic faced the speaker while the other was placed somewhere between the 2 loudspeakers. The unidirectional mic picked up both speech and background noise and only the background noise went into the omni-directional mic. Two microphone signals were recorded simultaneously and separately through a stereo PC sound blaster and saved on the recording PC in 16 bit 'wav' format.

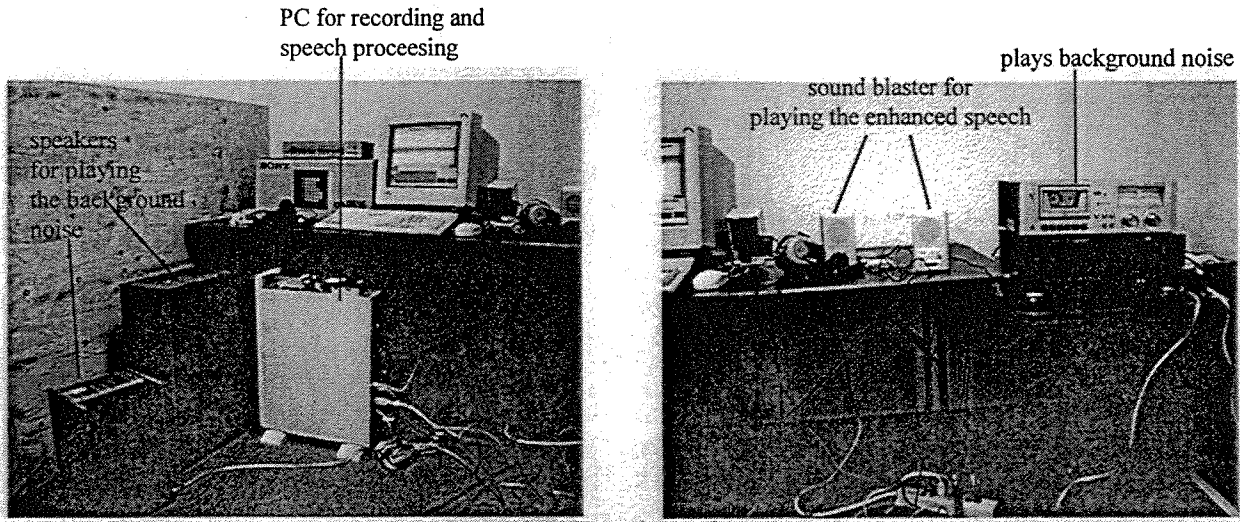


Fig. 2 Experimental setup.

Results

With the above the setup in mind we emulated three scenarios: 1) no speech, both speakers broadcasting a single tone signal; 2) no speech, both speakers broadcasting a signal that contained three frequencies; 3) a scenario where a pilot was speaking to the unidirectional microphone in a very noisy environment full of the helicopter noise. It should be noted that the improvement is not significant.

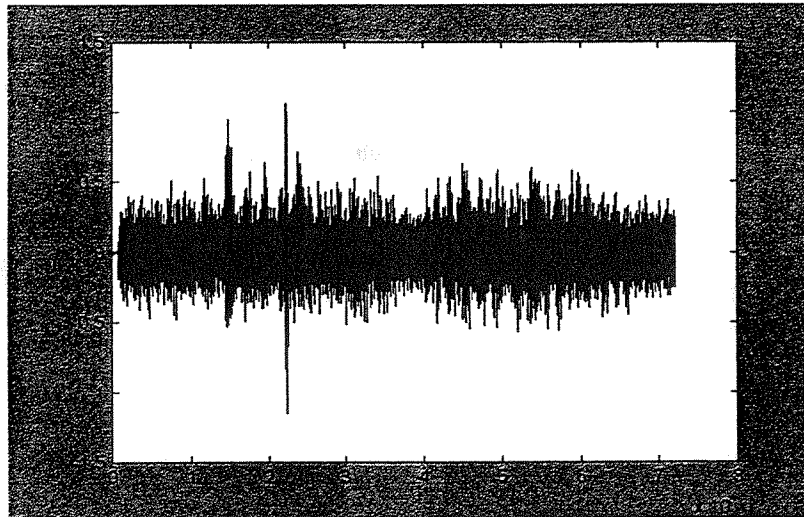


Fig. 3 Results of active noise cancellation for enhancing speech inside helicopter.

Approach 2: Noise elimination based on temporal filtering using one microphone

Figure 4 shows the block diagram of our approach. The key here is the speech detection algorithm which detects when there is speech or no speech. Experimental results show that this approach works very well.

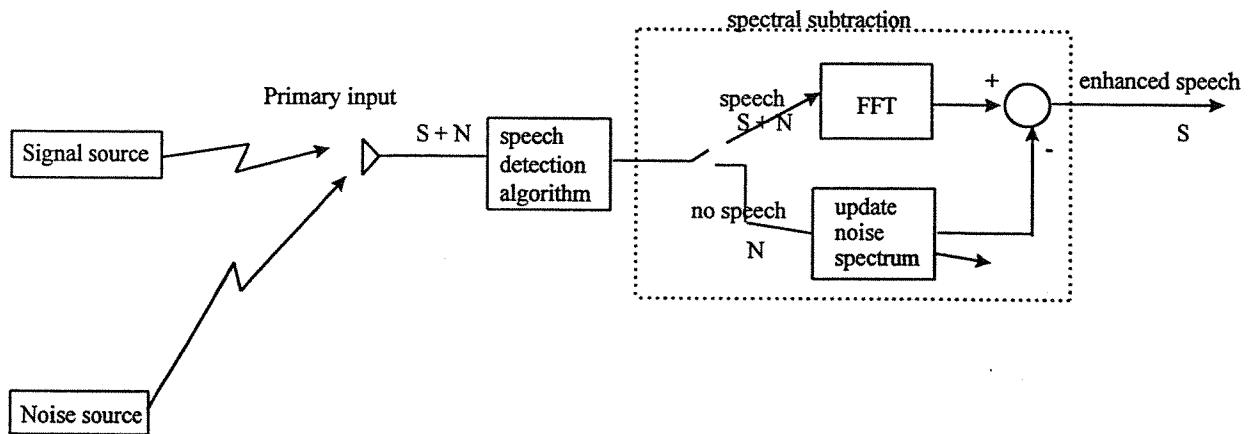


Fig. 4 One microphone approach to noise filtering.

2. Multifunction Phased Array

IAI has successfully applied PCA (Principal Component Analysis) and Fuzzy CMAC to adaptively adjust weights in frequency hopping multifunction phased arrays radar systems.

Fig. 5 shows a linear phased-array antenna. With the zeroth element having unity gain and the gain of the other elements weighted by complex weights w_1, w_2, \dots, w_M , the array response to a far-field source from a direction θ as shown in Fig. 5 is given by

$$D(z) = 1 + \sum_{m=1}^M w_m z^{-m} \quad (1)$$

where

$$z = \exp(j \frac{2\pi d \sin \theta}{\lambda}),$$

d is the antenna element spacing in the array,
 w_m 's are the weights that can be adaptively adjusted,
 λ is the wavelength of the incident signal.

By adaptively adjusting the weights of the antenna array, some appropriate antenna patterns can be formed according to the DOAs of the interference signals. For example, an antenna pattern is shown in Fig. 5. This antenna pattern has very sharp nulls at DOAs of 30 degrees and -45 degrees to reject interference signals arriving from 30 degrees and -45 degrees.

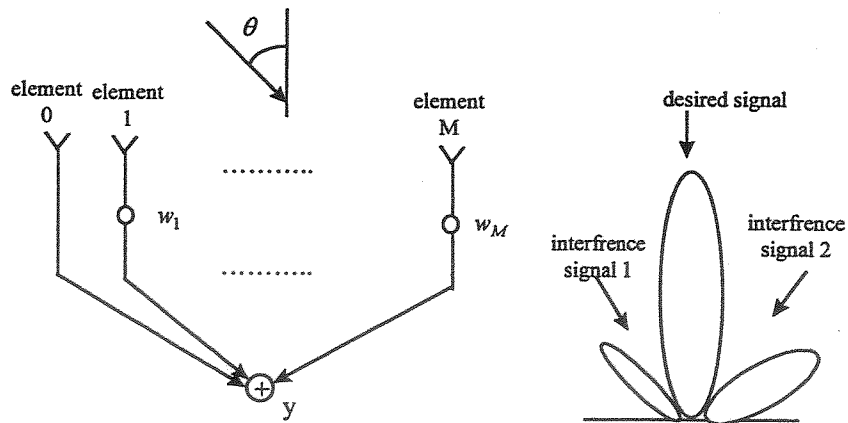


Fig. 5 Phased-array antenna and its effective antenna pattern.

We have developed a software system to implement the proposed interference cancellation system. The program consists of four parts: (a) Preprocessing to eliminate the desired signal (this is not necessary if the desired signal and the jammers are non-coherent); (b) adaptive principal component analysis to extract the principal eigenvectors which span the signal subspace; (c) a 1-dimensional search algorithm to locate the directions of arrival of the interference sources; (d) an off-line trained Fuzzy CMAC neural network which learns the nonlinear relationship between the DOAs, hopping frequencies, and the desired weights of the antenna array elements. One advantage of using Fuzzy CMAC is that the geometry of the phased array does not need to be linear, i.e. irregular array patterns can be dealt with. The overall scheme is shown in Fig. 6. The system makes use of two beamformers. The beamformer on the right is connected directly to the elements and is used to derive the array output signal. It is a slaved beamformer rather than the adaptive beamformer that would usually be expected. The beamformer on the left is our adaptive beamformer. The inputs to the adaptive beamformer consists of differences between array element outputs.

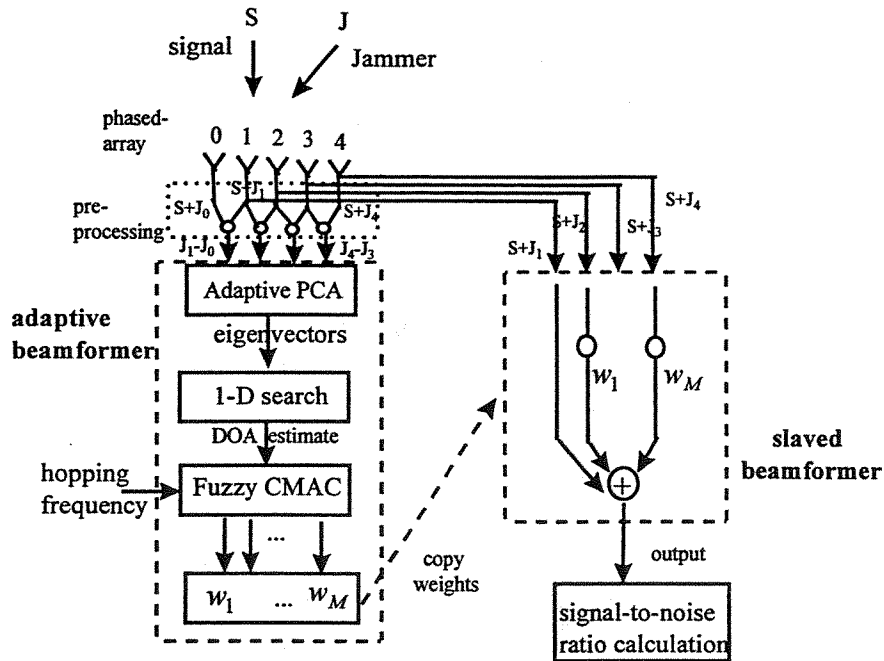


Fig. 6 Overall intelligent interference canceling system for frequency hopping applications.

Fig. 7 shows simulation results of a single stationary jammer with a frequency of 0.8 GHz. The desired signal comes from a DOA of 120 degrees. This DOA is rather arbitrary; other DOAs could have been chosen. The jammer is assumed to come from three different DOAs, i.e. 60, 90, and 150 degrees. There are three rows in Fig. 7 with each row corresponding to one case of DOA of jammer. There are 3 plots in each row of Fig. 7. The first plot is the signal-to-noise ratio (interference signal attenuation) versus time. The second plot shows the overall antenna pattern. The third plot shows the antenna gains versus angles of arrival.

From the first column of plots in Fig. 7, it can be seen that the proposed method can achieve high interference signal rejection within 0.1 ms whereas LMS method takes a long time to reach steady-state. Therefore, the results clearly demonstrate the proposed method achieves faster convergence than the LMS method. Most importantly, our proposed method can meet the requirement of frequency hopping systems for different DOAs, i.e. respond within 0.1 ms of the hopping period.

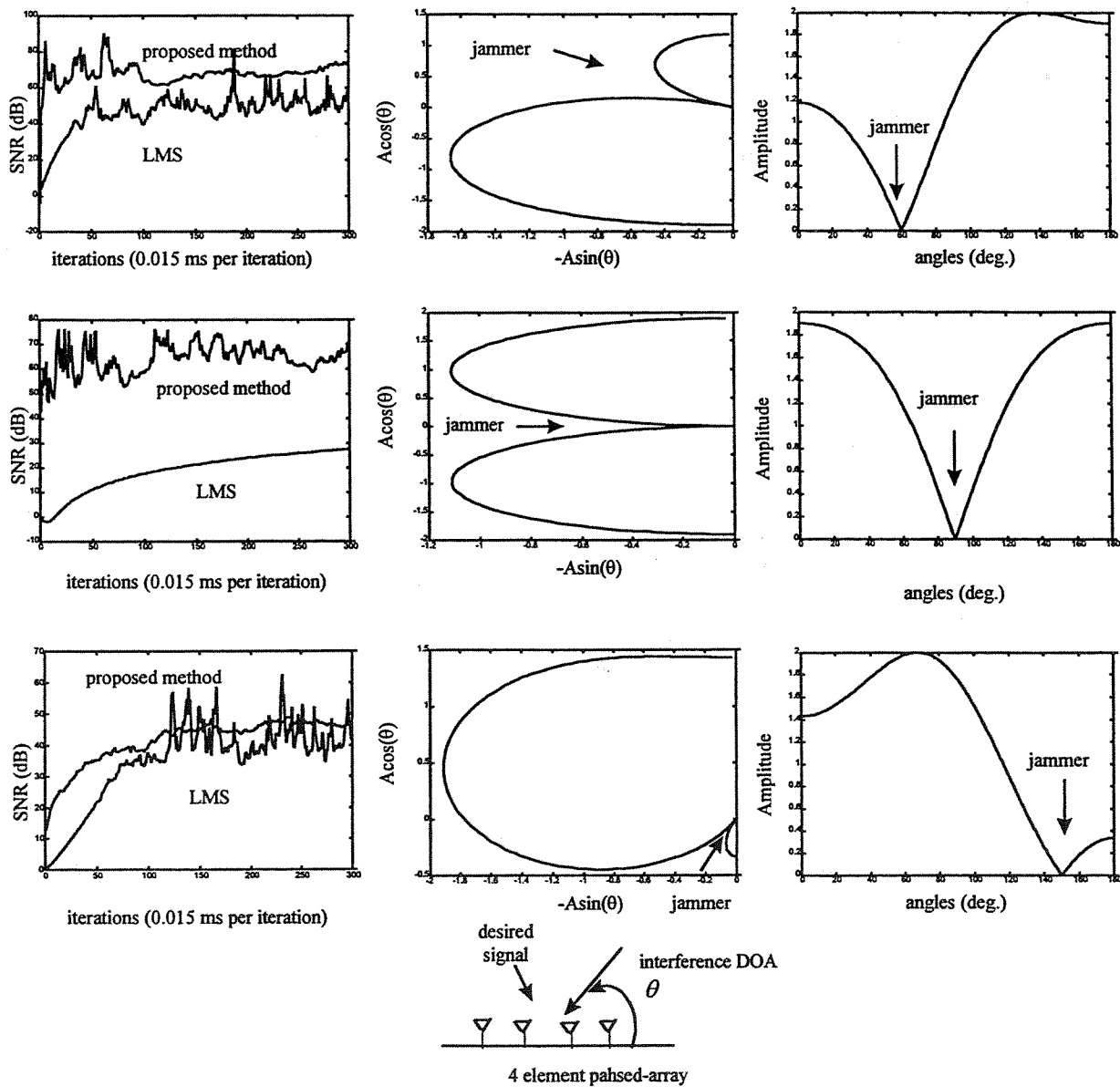


Fig. 7 Single stationary jammer with a frequency of 0.8 GHz. Three DOAs are included.

3. Face recognition using cameras

The bombing of the Trade Center in New York and the destruction of the Federal Building in Oklahoma City are manifestations of a dangerous problem in the United States and increasingly in the world. There are large segments of the population who are well armed, knowledgeable in causing destruction, and with sufficient anger and frustration that they view the deaths of hundreds of innocent victims without remorse. In this project IAI investigated a new approach to providing the authorities with better tools to identify and convict criminals. The innovation is to develop the next generation intelligent surveillance camera. The system will be capable of observation with a wide field view, detection of moving objects or persons within that field, zooming to capture high resolution images of those persons, and identifying persons based on facial features.

Imaging Hardware

Numerous commercially-available surveillance systems were considered before we undertook our own design. The most technically advanced systems were identified as "Cyberdome" from Kalatel, "SpeedDome" from Sensormatics, "AutoDome" from Beurlle Phillips, and the WV-CS604 from Panasonic. The real-time performance of

The key components of the image processing algorithms are also depicted in Fig. 8. The principal objective of the processing for the wide field camera is to find objects in motion. Motion can be detected in a number of different ways. For Phase I successive frames were differenced to isolate those regions that change from frame to frame.

First an image is captured, and the absolute value of the difference between the current image and the previous image is calculated. A median filter then operates on the differenced image. The median filter rejects "shot" noise or "salt and pepper" noise characterized by the presence or absence of isolated pixels or small groups of pixels commonly resulting from difference operations. Isolated pixels are replaced by the median value of the pixel group surrounding the isolated pixel. Next connected regions (or blobs) are isolated and labeled. Then each region is statistically analyzed to determine its potential to represent a person. The statistics used during Phase I were simple: the aspect ratio of the region as measured by the ratio of the major and minor axes, and the area of the region. During Phase II, each of these regions will be further screened by the face detection software. However, in Phase I, there was no high speed communication channel between the 2 PCs. Therefore, the centroid of all regions meeting the simple statistical screening is calculated, and the high resolution camera servoed to that location. The hi-res camera then begins the face detection and recognition process. The software used is an adaptation of "Facelt" from Visionics Corp.

Face Detection and Recognition

Detection: The first step in face recognition is to efficiently find the location of the face in the field of view. In a surveillance system, this entire process should take no more than 50-100 ms. The challenge comes from the fact that in our Phase II system the field of view of the camera is relatively large. In principle, one is computing the probability of a face at every location. Luckily faces on a background produce strong cues that can simplify the search. The most prominent are discontinuities in the spatial, temporal, and color domain. One way to reduce the dimensionality of the original search problem is to process the input image using the appropriate spatial, temporal and chromatic derivatives and to examine further only those regions where the discontinuities are significant. The precise form of these filters as well as the thresholds can be statistically derived from an ensemble of video segments and images of faces on a background. These cues can also be processed in parallel and in separate threads, and only when available. For example when the image is static or the camera is black and white, the system relies on the spatial cues only.

Alignment: The problem of ascertaining the existence of a head is not as difficult as precisely determining its position, size and pose. This requires detailed shape and feature detection. In regions where the detection module has given high probability of a head, the system integrates the edges into contour segments and template matches them against prototypical contour shapes of heads. If any exceeds a given threshold, the region is passed to the next stage which searches for the eyes and nose. If these features exist then the target is a face and is processed for alignment.

Precise alignment is a significant key to successful face recognition systems. This is because variability due to alignment errors dramatically affects the matching score of a face with the stored templates and leads to false entropy. Facelt estimates the alignment parameters (position, size, rotation and to a lesser degree pose) by reconciling three different types of information for added robustness. In Phase I we use the contour radius of curvature, the inter-eye distance, and a 3D model of the human head (Atick, Griffin and Redlich 1996) to find an independent estimate of those parameters.

Normalization: Once the parameters are estimated, the head is normalized by scaling, rotating and warping. One may also need to normalize with respect to lighting variability. For this we adopt two strategies. We compensate for large scale lighting variations on the face by first estimating the lighting pattern by relying on a 3D model of the face, but we also perform a 2D transform that creates a good degree of light invariance. This transform is a by-product of work on shape-from-shading (Atick, Griffin and Redlich 1996), and it allows us to construct the same output irrespective of how lighting is changed over the face, of course within a reasonable range.

Face print: Ultimately every face recognition system has an internal model for faces that is well suited for matching against the database. The representation used in Phase I is based on Local Feature Analysis (LFA). LFA is a mathematical technique for deriving local, topographic representations for a class of complex objects from an ensemble of examples (as in Principal Components Analysis). It was first introduced to machine vision by Atick and Redlich (1990) and was more recently formulated as a general mathematical theory for object representation by Penev and Atick (1996). It has the same mathematical complexity of Principal Components Analysis, but it results in a description of objects as a collection of local features. The method determines which local features are best matched to the class of

objects at hand and which particular features activate for a given object. For faces, LFA produces excellent feature detectors for eyes, noses, jaw-lines, cheek bones etc, and for a given face only a handful of key features activate. Which features activate changes from one face to another and hence adds valuable identity information.

Experimental Results

Using the hardware and software described above, we were able to achieve the objectives of motion detection using wide field vision, high-speed aiming of the narrow-field camera, and face detection and recognition with reasonable accuracy.

The Smart Camera was placed in the upper corner of a rectangular room opposite an entrance doorway. Experiments were conducted to detect persons entering the room and to determine the reliability of the system in determining their identity. The room is 16' in length x 11' in width x 8' high. The wide-field camera lens was selected to view all regions of the room and with the doorway in its center of view. Motion detection proved relatively straightforward. Frame differencing reliably revealed image regions in movement, and threshold values on the simple statistical measures of region aspect ratio and size reliably screened regions that could potentially represent persons.

The servoed mirror permitted us to step the narrow field camera to any location seen by the wide field camera five times per second. It is expected that with servo optimization we can achieve an order of magnitude increase in performance beyond that. Various modes of scanning were implemented. In one mode, the servo continuously tracks the region with the highest probability of being a person. In a second mode, the servo repetitively cycles through every object in motion. This assures that all persons in motion are subjects for face recognition. The dwell time on any region was selectable by the user. This mode effectively permitted any object to be tracked for a time sufficient to implement face recognition and then move on to other regions in motion. Unfortunately, in Phase I, there was no high speed communication channel between the two PCs to assure that sufficient time had been allotted for recognition.

A database of faces representing ten persons was constructed. Each person was imaged face-on from three slightly different perspectives. Face detection and recognition were achieved with an accuracy of approximately 60% for persons entering the room at modest velocity and walking toward the camera. For persons walking normal to the viewing axis of the camera and with the front of the face not exposed to the camera, detection and recognition accuracy declined considerably. Detection times were on the order of 0.1 sec; recognition times, on the range of 0.5-1.0 sec.

Digital Ink: A Familiar Idea with Technological Might!

Chris Kasabach, Chris Pacione, John Stivoric, Francine Gemperle, Dan Siewiorek

EDRC* at Carnegie Mellon University

Pittsburgh, PA 15213 USA

+1 412 268 7890

{kasabach, stivoric}@cmu.edu

<http://www.edrc.cmu.edu/design>

ABSTRACT

Digital Ink is a design research concept. Part design, part critique, it is the integration of current and future technologies into a mobile and socially familiar object. Digital ink is a sophisticated pen that allows people to take notes, sketch, and save the "physical" data they generate, digitally and automatically. It strives to turn mobile computing and interaction on it's head by turning the monitor into a piece of paper and the keyboard and mouse into the pen itself. It's designed so people can do things they normally do with any pen, but also fax, print, plan and correspond with others.

Keywords

design research, digital, pen, information, interaction, hand-drawn interface, mobile, future, concept, technology

INTRODUCTION

Consider the unhealthy posture the body assumes while working on a PC all day, or the abundance of cryptic products like VCRs, microwaves and ATM machines. We hover over technology, scratch our heads and push buttons. It seems that the technology industry often forgets that designed things affect experience. Instead, with each computer generation, the industry seems to make another capitulation to the preconditions of keyboard, screen, mouse and windows. How can we make computers move more smoothly with the momentum of everyday life – less about technology, and more about fulfilling the needs, habits and desires of people?

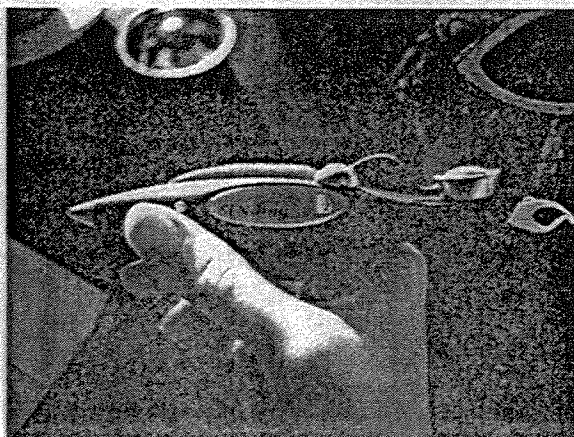
For the last six years the Engineering Design Research Center (EDRC) has been addressing this issue by designing and building mobile and wearable computers and inventing new interaction techniques for a variety of clients. One client, Intel, came to us in April 1996 asking us what computers could be like in five to ten years. This question opened the door on a real opportunity to change how we think about "being digital".

The feeling among members at Intel and our design group was that technology is moving fast enough to catch up to the things people say they really want to do with it. And

despite standards and numerous cases of technology lock-in (the QWERTY keyboard is a good example) there are opportunities to make our experiences with computers more sympathetic, and maybe even poetic.

MIGHTIER THAN THE SWORD

Everybody uses a pen. We carry them in our pockets and briefcases. Their usefulness lies in the mobility, simplicity, and immediacy with which they let us record thoughts or phone numbers on a handy napkin or even the back of a hand.



Digital Ink is a sophisticated writing tool that both understands people's handwriting, and allows them to turn any writing surface into a personalized interaction surface. By merging traditional tools such as pen and paper with the capabilities of an electronic notepad, modem and cellular phone, Digital Ink enables people to record, transmit and receive ideas from almost anywhere.

INTERACTION DETAILS

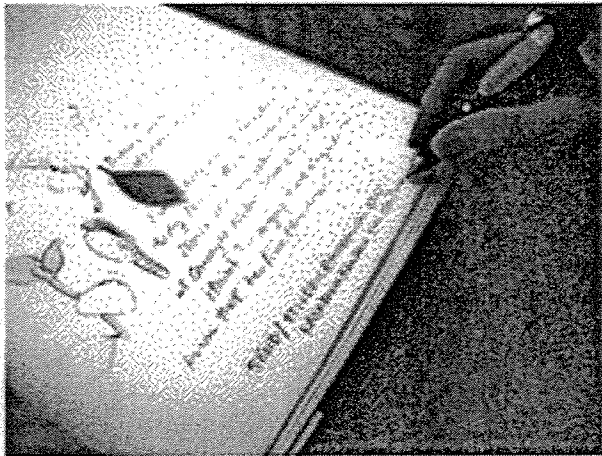
The interaction is designed around what people already do with pens – namely, make marks. All functionality is performed by writing or drawing with the pen. The only hardware function is the clicking between modes.

While on, the pen functions in two modes; record and command. The round mode button allows users to toggle between them as necessary.

In record mode, the pen is simply recording what is written and drawn. In command mode, the pen is reading written commands, finding keywords and performing those commands. For example, after writing a short note the user would click to command mode and write:

send to michelle@berlin.com-->

“Send to” invokes the command and prepares the pen to pay attention to whom. The “->” terminates the command and sends the message. The transmission’s status is displayed on the small elliptical LCD.



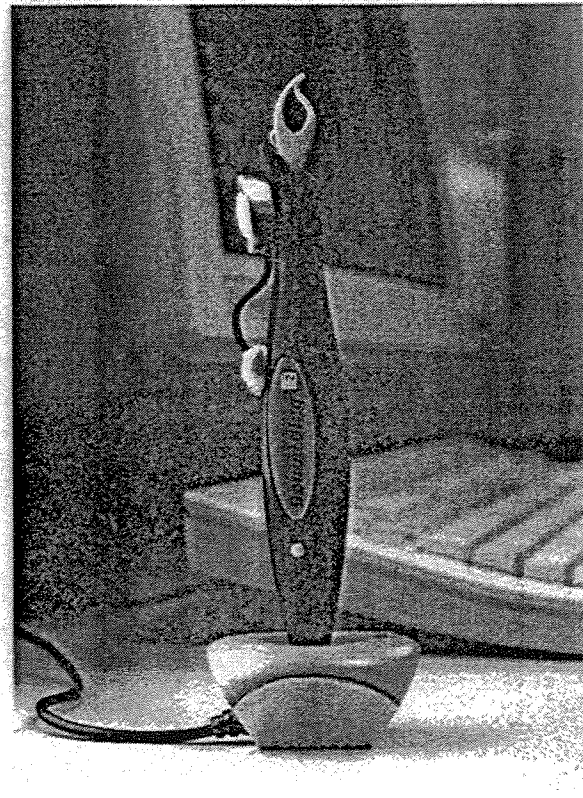
While in command mode the pen's elliptical screen slowly scrolls through the command words reminding the user of their options. The user can select between: send, save, read (e-mail and faxes serially), contacts, and download. Once a command word has been written and ‘read’ by the pen, the screen displays the progress of the command.

The same cellular components that allow the pen to send and receive e-mail allow it to function as a telephone. This is done with the addition of an external speaker & microphone attachment. This attachment allows the pen to receive voice input in both the record and command modes.

The small flame-like pen clip covers Digital Ink's pen tip so the pen can be carried in a pocket. The Clip also clips onto the writing surface, assisting the pen's ability to understand page boundaries and writing locations. Micro-accelerometers, in combination with software algorithms, measure the angles and direction of lines put on the writing surface.

Such accurate line recognition allows Digital Ink to understand the meaning of basic shapes and their relative positions on a writing surface. For example, the pen can understand a drawing of a calculator as a calculator. Draw a tool when you need it!

The small digital "ink well" connects to any computer and serves as home to Digital Ink. It also serves as the pen's recharging station and downloading port.



NEXT STEPS

We still have many challenges ahead. For example, today a production version of Digital Ink would be fairly power intensive, and there are still many interaction and technology issues to be addressed. We are working on it with Intel. The first working prototype of the pen, currently under construction, will be able to capture words and drawings as they appear on a page.

ENABLING TECHNOLOGIES

Current and future technologies that will support this concept are described in the Digital Ink video.

ACKNOWLEDGMENTS

We would like to thank Patrick Mitchell and Michael O'Connor, Intel; Dick Urban, DARPA; Dan Siewiorek, EDRC at Carnegie Mellon and Richard Martin, Robotics Institute at Carnegie Mellon; Len Bass, Craig Vogel, Dave Aliberti, and Nora Siewiorek.

*The EDRC resides within the Institute for Complex Engineered Systems (ICES).

Foldable Computing: Designing a computer that adapts to your information needs

Chris Kasabach, Chris Pacione, John Stivoric, Francine Gemperle, Dan Siewiorek

EDRC* at Carnegie Mellon University

Pittsburgh, PA 15213 USA

+1 412 268 7890

{kasabach, stivoric}@cmu.edu

<http://www.edrc.cmu.edu/design>

ABSTRACT

The Foldable Computer is a design research concept that physically adapts to the information needs of its user. Starting at about the size of a rectangular wallet, this computer system unfolds three times to become the size of a medium size notepad, a paper back book, and finally a high-resolution 8.5" x 14" display.

Keywords

foldable computer, rigid polymer LCD panels, piezo-electric motors, design research, user-centered

INTRODUCTION

The foldable computer was developed for the growing group of professionals who do most of their work in the field or on the move and need a great deal of information with them. Specifically, the Foldable Computer addresses the problem of interacting with both simple and complex data all with the same mobile computer system.

Currently, there are a variety of "one size fits all" products on the market, like notebook PC's, sub-notebook PC's, and now products like the Sharp Zaurus or the HP Omni-pro. These last two products offer not only PDA functionality but features like internet access and e-mail correspondence in the palm of your hand.

Although useful, all of the products mentioned above are simply miniaturizations of larger products and become difficult to use when moving around and interacting with complex information like maps, large blueprints, schematics, spreadsheets, and websites.

THE SOLUTION

Our solution is a device that can be "re-formed" to fit the user's task. In its smallest configuration the foldable computer fits in a shirt or pants pocket. It provides similar functionality to that of other small PDAs like the 3Com PalmPilot. Unfolding the Foldable Computer once doubles

its size, providing the ability to take notes and respond to email. Unfolding the computer a second time transforms it into a book form factor, suitable for web browsing and other electronic book documents.

Completely unfolded, the display screen is 8.5" x 14", and suited for large, complex information. Moreover, at this size, the computer can be attached to a keyboard, powered from a wall outlet, and used like a traditional desktop computer.

THE TECHNOLOGY

The Foldable Computer takes advantage of newly emerging manufacturing technologies. These include advances in flexible polymer LCD displays and system-on-panel component assembly.

In its smallest form, the Foldable Computer system is basically eight, VGA resolution, rigid polymer LCD panels, connected and stacked on top of one another. Protecting these panels is a leather wrap. Unsnapping and folding back this wrap turns the computer on.

As the LCD panels are unfolded, the computer increases its display size, opening all the way up to a usable 8.5" x 14" surface area at a 1280 x 1920 pixel resolution. The display surface is scratch-proof and electrically active only when engaged by the user's pen or finger touch.

Flexible interconnects, sandwiched between piezo-electric motors allow the folding and rigidization of the panels as the computer is unfolded. The computer recognizes when panels are opened past 165 degrees and activates the piezo-electric motors to snap the panels into a rigid position. This alleviates the user from having to hold the computer taught while in use. With each fold, small embedded magnets in the panels create an audible click, notifying the user that the panels will hold in place.

Providing the computing power of the foldable computer are metallic modules embedded in the back of each LCD panel. Each module contains a single computer function

such as central processing, solid state memory, digital cellular communications, speech recognition, global positioning, and power. A photo-voltaic ink for collecting solar and artificial light is coated on the back of each module to recharge the batteries.

NEXT STEPS

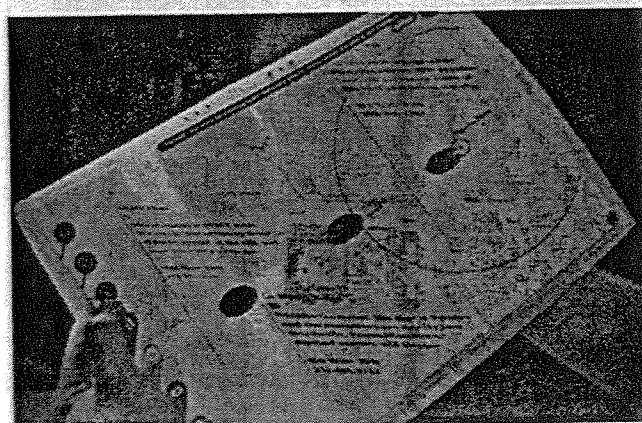
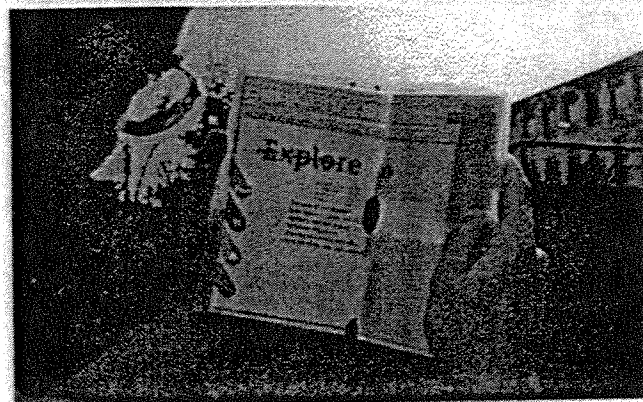
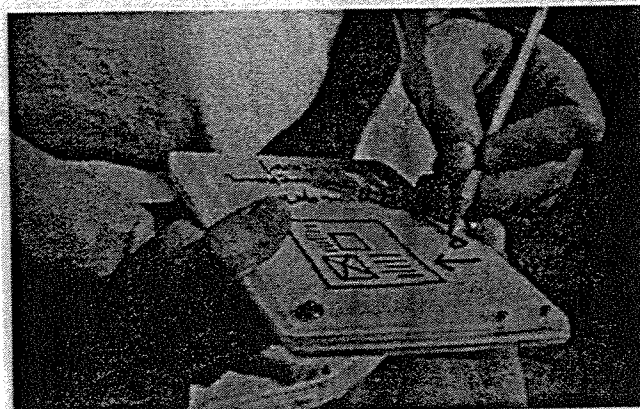
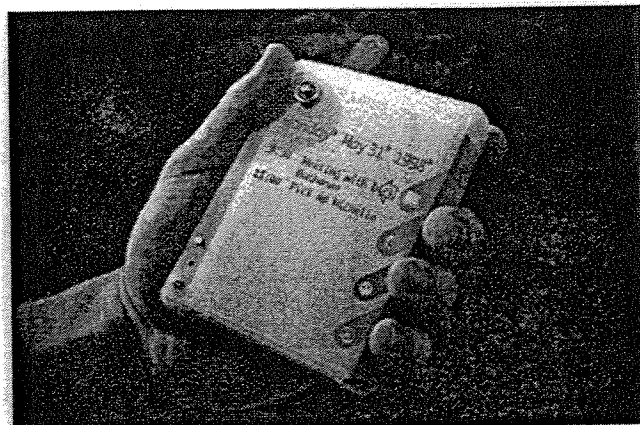
The Foldable Computer is a concept developed through a challenge from Intel to create new and improved mobile computers that could exist in five to 10 years. We are working with Intel, researchers at Carnegie Mellon and several other companies to make this concept a reality.

ACKNOWLEDGMENTS

We wish to thank Patrick Mitchell and Michael O'Connor, Intel Corporation; Dick Urban, DARPA; Richard Martin, Robotics Institute at Carnegie Mellon; Len Bass, Craig Vogel, Dave Aliberti, and Nora Siewiorek.

*The EDRC resides within the Institute for Complex Engineered Systems (ICES).

©1998 copyright on this material is held by the authors



PROMERA: A Computer, Projector and Camera all in one.

Chris Kasabach, Chris Pacione, John Stivoric, Francine Gemperle, Dan Siewiorek

EDRC* at Carnegie Mellon University

Pittsburgh, PA 15213 USA

+1 412 268 7890

{kasabach, stivoric}@cmu.edu

<http://www.edrc.cmu.edu/design>

ABSTRACT

PROMERA is a design research concept based on the integration of three familiar technologies; a pen-based computer, a projector, and a video camera. Weaving these technologies together with new but intuitive interaction techniques we have developed a multi-function, hand-held personal computer.

Keywords

RGB semiconductor lasers, camera/projection toggle, squeeze, liquid crystal display (LCD), flexible pen, gyroscopic navigation, design research, user-centered

INTRODUCTION

At the close of the century the technology industries have created a powerful, yet for now, unharnessed accomplishment. They have developed the wireless and web based infrastructure capable of obtaining and interacting with virtually any information, any place and at any time.

Medical treatments, bus schedules, recipes, stock transactions, help groups, currency conversions, e-mail, road maps, dictionaries, catalogs, addresses. They are all floating out there, accessible, not always in the form we desire, but present nonetheless.

Not surprisingly, this daunting accomplishment has some quirks. Most noticeable is that our new and massive information capability has come faster than the portable vessels and devices required to make the information inviting to use.

For now we have laptop computers and Personal Digital Assistants (PDA's), but these have limits. The former lets us do big and powerful things but is in itself big and unusable on the move. The problem with the latter is it is small and does only small things.

WHERE PROMERA FITS IN

PROMERA is a palm-held, donut-size computer that affords all the capabilities of a laptop computer but with interaction methods that allow it to be small itself. Moreover, it has several other interaction techniques that extend its ability beyond conventional computers.

It is small enough to fit in a pants pocket but can display information at large sizes.

HOW IT WORKS

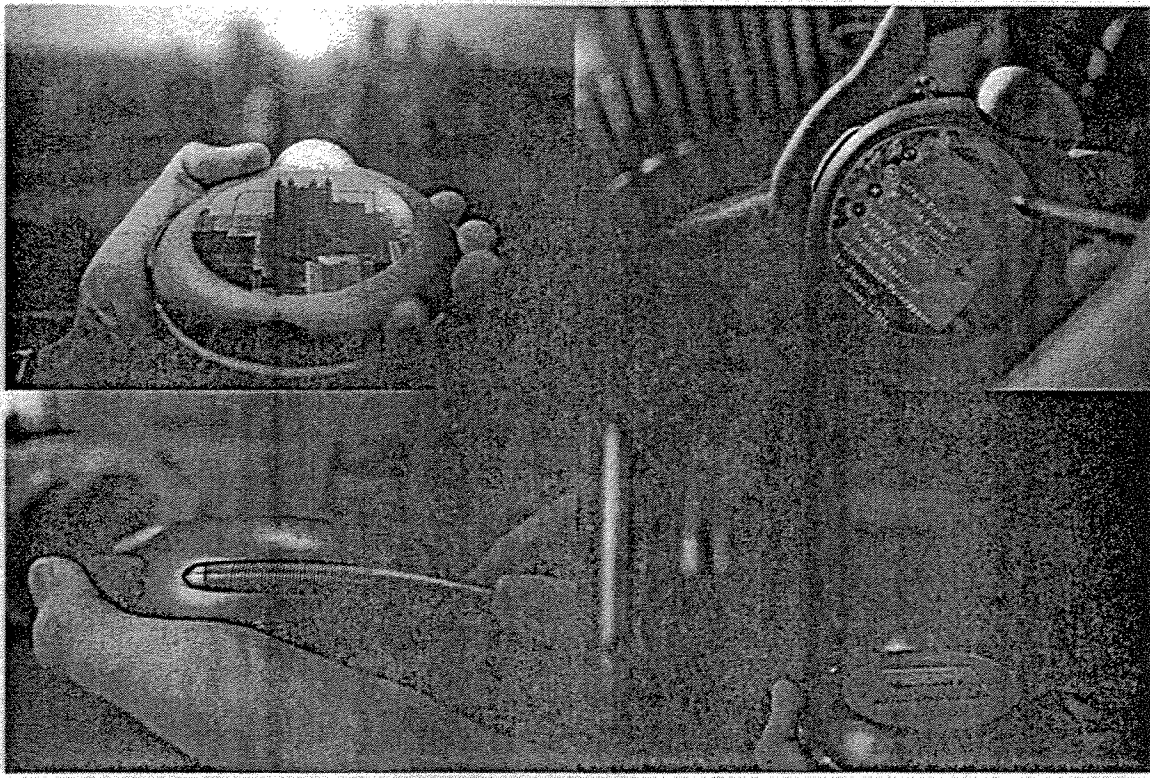
PROMERA has two means of gathering information – cellular components that allow it to collect information wirelessly from servers, and an integrated still/video camera for gathering images. In camera mode PROMERA has a 6x zooming capability and large soft leather pads on its sides to zoom images in and out. The harder PROMERA is squeezed the closer the image becomes. Releasing pressure on the pads “releases” the image back out. This system makes interaction with the information much more intimate and direct than flipping small levers or switches for zooming.

Once information is gathered, PROMERA employs several interaction techniques to view and manipulate it. First, information gathered into PROMERA can be viewed on its round liquid crystal display (LCD). Or, for larger viewing, information on the LCD can be projected back through the camera lens. This is done using a mechanism much like a single lens reflex (SLR) camera uses. In camera mode PROMERA's CCD chip receives still and video images. In projection mode the CCD chip folds down and an array of RGB semiconductor lasers are exposed. These small lasers project the information on PROMERA's display at high-intensities and low-power on any flat surface. These are the same lasers used in wand-like light pointers.

The benefit of projection is that information can be viewed larger than desktop computer screens and is unconstrained by rectangular formats. Information can be displayed in the shape it makes most sense to view it.

PROMERA has a flexible pen that stores around its perimeter. Once removed, the pen straightens and is used for selecting items on PROMERA's display, writing commands such as website searches, drawing on and annotating pictures taken with the camera, and also for

navigation. This last function is used in projection mode. Sliding the pen on PROMERA's display creates a shadow across the same place on the projected image. Similar to a mouse and monitor it is possible to look up at the image and interact non-visually with the input device.



As a last form of interaction PROMERA has gyroscopic components that permit information on the round LCD to be scrolled and expanded. For example an extensive list that is longer than PROMERA's screen can be viewed by tilting PROMERA in the direction you want the list to scroll. Once the information comes into view, simply tilt PROMERA back in the other direction or tap the "hold" command.

NEXT STEPS

PROMERA is a concept developed through a challenge from Intel to create new and improved mobile computers that could exist in five to 10 years. In truth, most of the technology necessary for PROMERA to work, works today. However, the size of these technologies, their

power consumption and thermal properties have yet to be optimized for such a device. We are working with Intel and other researchers at Carnegie Mellon to make this concept a reality.

ACKNOWLEDGMENTS

We wish to thank Patrick Mitchell and Michael O'Connor, Intel; Dick Urban, DARPA; Richard Martin, Robotics Institute at Carnegie Mellon; Len Bass, Craig Vogel, Dave Aliberti, and Nora Siewiorek.

*The EDRC resides within the Institute for Complex Engineered Systems (ICES).

©1998 copyright on this material is held by the authors

THE SHADOW: A PERSONAL EXPERIENCE CAPTURE SYSTEM

"Every bit of information we create is useful to someone."

James A. Landay, Mark Newman, Jason Hong

Computer Science Division
University of California at Berkeley
Berkeley, CA 94720-1776
+1 (510) 643-3043
landay@cs.berkeley.edu

ABSTRACT

We propose to build a "personal experience capture system," which unobtrusively follows us wherever we go and knows everything that we do. This Shadow will be able to create a precise record of all of our actions and experiences. High level semantic information will also be captured. Individuals can use this data to assist in recalling information. By collectively networking the information, we can share our knowledge and experiences with one another. This paper outlines our ideas and the research problems that must be solved to achieve this vision.

1. INTRODUCTION

The system we propose is a "personal experience capture system", which unobtrusively follows us wherever we go and knows everything that we do. This *Shadow* will make use of computing devices ubiquitously embedded in our environment to create a precise record of all of our actions and experiences. These records can include audio, video, physical location, and time data, as well as other formats, such as logs of all of our correspondence and documents we have edited. High level semantic information, such as where we have visited, whom we have met, what was said by whom when, what steps were performed in accomplishing a task, will also be captured. Individuals can use this data to assist in recalling information. Furthermore, by collectively networking the information, we can share our knowledge and experiences with one another.

A Shadow is a disembodied process that follows a particular user and is able to make use of whatever devices and network services it encounters. In the near term, location and identification of users will be accomplished by user-carried Personal Digital Assistants (PDAs) in a seamlessly connected environment. The concept is not tied to this means of identification. Image and voice recognition, when mature, could be sufficient to identify and locate a user within a smart environment.

The basic concept behind a Shadow can be traced to Vannevar Bush [1], who noted that "A record ... must be continuously extended, it must be stored, and above all it must be consulted."

Gordon Bell has proposed a Guardian Angel that can "retrieve everything we hear, read, and see" [2]. Dan Olsen has also suggested a similar system that we would wear [3]. In effect, a Shadow would be what Don Norman calls a cognitive artifact, or a tool that aids the mind. Norman writes, "the technology of artifacts is essential for growth in human knowledge and mental capabilities" [4].

2. APPLICATIONS AND BENEFITS

2.1. Journal / Notebook / Manual

A Shadow will be extremely useful whenever we need to recall precise details of our past. While people are good at many things, recalling specific details with complete accuracy is not one of them. A Shadow can assist us by allowing us to review important details of our past experiences.

One application in this domain is the Journal. The *Journal* allows us to browse or search the data collected as a record of our personal experience. We can record and recall ideas, conversations, promises, and appointments by browsing the journal. The *Notebook* is a more active version of the journal, in which the data passively collected by the Shadow is augmented by data explicitly supplied by the user. The Notebook is specialized towards collecting and displaying information in the interest of facilitating document creation, not just information browsing and searching.

Another application in this domain is the Manual. A *Manual* assists us in remembering how we successfully accomplished something in the past, which will not only improve our performance but prevent us from wasting time. The Manual can be extremely useful in an educational environment. Although there are many things we have learned in our past, it is difficult to remember the specific steps to do something unless it is done on a regular basis. The Manual allows us to distill the essence of what we have learned, so that if it is necessary, it can be quickly recounted.

2.2. Personal Assistant

An agent could also use the data collected by our Shadow to learn our preferences and interests, and even act on our behalf. The Shadow Agent could negotiate with resources in the environment to ensure that our needs and desires are met. A simple example is a *Personal Assistant* that negotiates with a smart environment to maintain an optimal room temperature for a given user. The Agent knows the preferences of the user and tries to obtain the user's desired temperature. Meanwhile the environment attempts to balance the requests of multiple users' Agents and arrive at the best solution.

Another use of the Personal Assistant would be to suggest information relevant to the current situation. This information can be given to us directly, as in Remembrance Agents [5], or given to us by subtle cues in our environment, as in calm computing [6, 7].

2.3. Personal Manager

A Shadow will also be useful whenever we need metrics on some aspect of our behavior. Metrics allow us to predict future behavior from past behavior, aid us in planning for the future, and assist us in pinpointing the source of a problem. However, people are not very good at recording metrics because it takes enormous discipline to record metrics consistently and accurately, and because recording metrics usually distracts from the task at hand.

One useful application in this domain is the *Time Manager*. The Time Manager allows us to see how we spend our time at any granularity, whether it be hours, days, weeks, or years. This could be used to see if we are managing our time well. The Time Manager can also assist us in planning. If we can determine how long we spent on previous projects, the Time Manager can assist us in planning how long new, similar projects will take. One can imagine a whole suite of applications, such as a Money Manager, an Exercise Manager, and even a Food Manager. If combined with the Personal Assistant, the Personal Manager could analyze one's behavior and suggest improvements.

2.4. Group Shadow

A *Group Shadow* is an extension of the Shadow idea to a group of people. All of the applications aimed for a single person can also be used for a group. For example, a *Group Journal* would record all of the interesting events for a group. A *Group Manual* would enable the sharing of "how-to" information among group members by simply tracking the processes of the experts. A *Group Manager* could capture metrics on a group, assist in pinpointing problems the entire group encounters, as well as assist in planning group projects. Essentially, a Group Shadow could establish a Group Memory, so that nothing would ever be lost.

For example, suppose a well-experienced system administrator leaves a company and is replaced by an inexperienced one. By using the captured know-how of the previous system administrator, the new one should be able to do anything the

previous one could (albeit more slowly). While not a perfect replacement, the Group Shadow could help bring the new system administrator up to speed on how things are setup and how things are specifically done.

Group characteristics could also be observed from aggregated Shadow information. Information such as traffic patterns, space and resource usage, group preferences and interests could be derived from sets of Shadow-collected anonymous data.

The Bootstrap Institute, founded and directed by Doug Engelbart, uses the term "Collective IQ" to describe how quickly a group can "leverage its collective memory, perception, planning, reasoning, foresight, and experience into applicable knowledge" [8]. They note that a key factor in this Collective IQ is the quality and utility of the group's knowledge repository. A Group Shadow could greatly aid in this endeavor.

3. OBSTACLES AND APPROACHES

3.1. Infrastructure / Heterogeneous Devices

For the purposes of a Shadow, we assume seamless, constant connectivity for users in all environments. Furthermore, we assume that it is easy to locate and identify users. Work on these issues is already underway here at Berkeley in the Daedalus group [9, 10]. We do not wish to duplicate their work but to extend it in several ways.

First, even assuming seamless connectivity, we still need to develop a semantics with which to express capabilities of devices and resources so that these component resources can be composed into complex, intelligent environments. HP's JetSend protocol [11] is one step towards addressing this problem. As time progresses, however, the number of computing devices per person will increase. Adding new devices, or more abstractly, new resources to an environment thus needs to be a painless operation, i.e. there should be no special configuration necessary to incorporate a new resource into an existing environment.

Similarly, there should be no special configuration necessary to incorporate new interaction devices into an existing environment. User interfaces to Smart Spaces should be composable "on the fly," on heterogeneous client devices, from desktops to laptops to PDAs to smart phones to pagers to as yet unheard of interaction devices. Defining the description semantics that will allow arbitrary devices and resources to learn of each other's capabilities and "do the right thing" is a necessary step to make the existing Daedalus infrastructure "smart."

Lastly, our environment should be composed of both physical and virtual spaces for several reasons. First, it allows an environment to be a combination of both physical objects and virtual objects. An example of a virtual object could be software that does additional post-processing on captured information. Second, it allows us to create spaces of any granularity, from desk-sized to room-sized to building-sized spaces to city-sized spaces. Third, Smart Spaces can be composed together to form larger, virtual Smart Spaces. For example, all the divisions of a company can

have their own Smart Space, but all of these individual Smart Spaces can also be combined into a larger, virtual Smart Space. This virtual Smart Space could have all the state, behavior, and knowledge base of its components. One approach to creating a virtual space is the Jupiter System [12], a multimedia network place that supports virtual objects and virtual tools. WorldBoard [13] has a novel approach in augmenting physical space by allowing every square meter on the planet to be marked up with virtual data.

One challenge in creating a Shadow in such an environment lies in the continuous capture of interesting information. A Shadow must always be with us and must always be capturing useful information. The Factoid project [14] accomplishes this by placing all of the capture in a mobile device. However, a Shadow only requires a mobile component for when the user is outside of a Smart Space. Ideally, the Smart Space could capture all relevant data whether or not you have a mobile device.

We can create a hybrid system using both mobile devices and Smart Spaces. Since the mobile device is always with us, we can always capture certain types of information. In addition, when we enter a Smart Space, the capture capabilities of the mobile device are merged with that of the Smart Space. The mobile device can seamlessly act in conjunction with other devices to capture richer forms of information.

For example, if we are travelling and happen to be in a place that does not have any capture devices installed, then the information that will be captured is limited by the capabilities of our PDA. However, if we are in a classroom that has audio and video recording devices installed, these devices can join together with each of the students' PDAs to capture aural and visual information, any information the instructor sends to us, as well as any notes taken in class on the PDA.

3.2. Context Awareness

Another challenge in creating a Shadow is in context awareness. Certain kinds of information will be very easy for a Shadow to capture, such as position and distance. However, these kinds of information by themselves are neither interesting nor useful. Audio and video can also be captured, but besides being difficult to search through (addressed below), the computer will still need a way to capture the essence of what we are doing.

For example, while a Shadow can record the audio and video of us working on our car, we really want the Shadow to know we are working on our car, changing the oil, and adding brake fluid. If it does not know what we are doing, it will be difficult to search for when we last changed our oil. It will also be difficult to have the Shadow assist us or predict our actions.

Again, it will be relatively easy for the computer to capture certain kinds of context, such as the current time and who is nearby. Much more difficult will be for the computer to capture and infer interesting events, such as studying for a test or designing a new system. Figuring out the who, where, and when can be done with existing technology, but much more difficult (and much more useful) is figuring out the how, what, and why.

One way to start on this problem is to capture information that already has some context known to the computer, in this case, computer applications. We can easily capture three pieces of information: the name of the application, the name of the file(s) worked on, and the time spent on each file. This is a simple way of prototyping some of the basic functionality that will be needed later on.

Another way to approach this problem is to use pre-existing context, such as calendars and to do lists. For example, if a person has scheduled an event on her calendar, a computer could assume that all captured material could be indexed under that event.

A third approach, applicable to restricted domains, would involve machine learning. A person could create explicit categories, such as "recent" and "school", and explicitly place items under these categories. The computer could use pattern matching techniques to determine what similarities items placed in the same category have, and eventually be able to predict what categories an item should be placed under. This approach is used for email messages in Re:Agent [15].

3.3. Presentation / Visualization of Data

A Shadow will collect huge amounts of raw data in many different formats, including video, audio, text, and images. This data will need to be processed and presented in intelligent ways in order to be useful. The key will be to conceive of applications like those outlined above and develop presentation techniques appropriate for specific applications. These presentation and visualization tools should also be designed to exploit the smart environment, as described in Application Development below.

Using web search engines may be an initial way to start, since the web also contains massive amounts of information. One key difference is that while information on the web is extremely diverse and disparate, a person's captured information need not be. The information can be indexed by various contexts (such as people, place, or time), and can be presented to the user in familiar formats, such as calendars, photo albums, diaries, and to-do lists.

Eldridge, Lamming, and Flynn showed that a Video Diary [16] helped people remember more than a written diary did. They also discovered that people and objects were often the cues used for recall.

Lamming and Flynn have also created the Forget-me-not system, a memory prosthesis implemented on the ParcTab PDA [17]. The Forget-me-not was designed to help overcome everyday memory problems by exploiting episodic memory. People naturally organize events in memory by placing it in the current context, such as where the event happened, who was around, and so on. The Forget-me-not would capture some of these events and allow users to interactively filter the system to view past events.

Schank [18] hypothesizes that stories are an organized and compact form of information storage. Norman [4] also anecdotally notes that people often make decisions based on

stories, not hard facts. An intriguing line of research would be to see if certain mechanisms improve certain kinds of memory.

3.4. Application Development

In Smart Spaces, the "computing environment" is no longer an abstraction to describe the invisible structures inside the machine on your desk (or the network in your walls), but it is nearly synonymous with the actual environment. In such a world, how will applications be developed? User interface components, nowadays almost always presented on a single machine and a single display, can be distributed across multiple specialized devices. Data input and output functions can, and almost certainly will, be divided among several machines.

In the component-based infrastructure described at the beginning of this section, device capabilities that are known at runtime can be dynamically exploited by applications that may have been written with no knowledge of the conditions under which they would ultimately be run. What abstractions can we develop to support programming in such a world?

3.5. Agents

Creating a system that can analyze patterns and trends will be another challenge. Advances in data mining and pattern matching techniques will be needed in order to accomplish this. A related problem is predicting a user's action, suggesting courses of action, and in some cases, executing simple actions. This problem is essentially a generalization of agent software.

SUMMARY

This paper has outlined a proposal for a "personal experience capture system." The Shadow will be able to create a precise record of all of our actions and experiences. By collectively networking the information, we can share our knowledge and experiences with one another. In this paper we have outlined our ideas and the research problems that must be solved to achieve this vision.

REFERENCES

1. Bush, Vannevar. As we may think. <http://www.isg.sfu.ca/~duchier/misc/vbush/vbush.txt>
2. Bell, Gordon. The Body Electric. *CACM* February 1997, Vol. 40, Number 2.
3. Olsen, Dan. Interacting in Chaos. Invited talk given at Intelligent User Interfaces 1998, San Francisco.
4. Norman, Donald A. *Things That Make Us Smart*. Addison-Wesley. 1993.
5. Remembrance Agents. <http://rhodes.www.media.mit.edu/people/rhodes/RA/>
6. Weiser, Mark. "The Computer for the Twenty-First Century." *Scientific American*, pp. 94-10, September

1991. <http://www.ubiq.com/hypertext/weiser/SciAmDraft3.htm>
7. Weiser, Mark. "Does Ubiquitous Computing Need Interface Agents? No." Invited talk given at MIT Media Lab Symposium on User Interface Agents, October 1992. <http://www.ubiq.com/hypertext/weiser/Agents.ps>
8. Bootstrap Institute. <http://www.bootstrap.org/vision.html>
9. Daedalus Project. <http://daedalus.cs.berkeley.edu/>
10. Hodes, T. D., and Katz, R. H. Composable Ad-hoc Location-based Services for Heterogeneous Mobile Clients. <http://daedalus.cs.berkeley.edu/publications/services-WINET.ps.gz>
11. JetSend Protocol. <http://www.jetsend.hp.com/>
12. Curtis, P., Dixon, M., Frederick, R., and Nichols, D. The Jupiter Audio/Video Architecture: Secure Multimedia in Network Places. *Proceedings of Multimedia 95*, ACM Press. San Francisco, CA.
13. Spohrer, Jim. WorldBoard. <http://www.almaden.ibm.com/almaden/npuc97/1997/spohrer.htm> <http://wtlinux.wisdomtools.com/wb/title.html>
14. Mayo, Bob. Factoid. <http://www.research.digital.com/wrl/projects/Factoid/index.html>
15. Boone, Gary. Re:Agent, An Intelligent Email Management Agent. <http://www.cc.gatech.edu/grads/b/Gary.N.Boone/reagent/reagent.html>
16. Eldridge, M., Lamming, M., and Flynn, M. Does a Video Diary Help Recall? <http://www.rxc.xerox.com/publis/cam-trs/html/epc-1991-124.htm>
17. Lamming, M., and Flynn, M. "Forget-me-not" - Intimate Computing in Support of Human Memory <http://www.rxc.xerox.com/publis/cam-trs/html/epc-1994-103.htm>
18. Schank, R. *Tell Me a Story : A New Look at Real and Artificial Memory*. Northwestern University Press. 1995.



An Employee-Owned Company

Science Applications International Corporation

Dynamic Network Computing: A Vision of the Next Information Processing Paradigm

*Submitted to DARPA/NIST Smart Spaces Workshop
July, 1998*

Title:	Dynamic Network Computing
Topic:	Smart Spaces
Name:	Richard A. Luhrs
Phone:	703-448-6558
E-mail:	Richard.A.Luhrs@cpmx.saic.com
Company:	Science Applications International Corporation
Address:	1710 Goodridge Drive, MS 2-8-1 McLean, VA 22102

Dynamic Network Computing

A Vision of the Next Information Processing Paradigm

1. Abstract

Recent technology developments are pointing the way to another fundamental shift in information technology. The current state of the information technology involves many desktop and server processors interacting on a large, heterogeneous network from a set of (mostly) static connection ports. Users must be near one of these fixed assets to access their information domain, or at least be able to plug their portable "laptop" into a fixed telephone port. By contrast, the concept described herein involves the emergence of dynamically reconfiguring networks providing a range of services to a mobile clientele possessing devices with limited processing and user interaction capabilities. To achieve their full utility, these portable devices will augment their capability by tapping the resource reserves of the information infrastructure. Their users will always be connected, taking advantage of the resources present in their current local environment and adapting as the user moves from one "infosphere" to another. Because both the network topology and allocation of processing tasks will be constantly in flux, we call this paradigm "Dynamic Network Computing", or DNC.

The technologies required to realize DNC are largely either present today or under development. However, a few technological barriers of modest dimension remain, and if the vision is to be realized according to some constructive plan (as opposed to being subject to the chaotic forces of the marketplace), then a program must be identified and orchestrated that will:

- motivate the research community with a common vision, providing consistent direction to research and development efforts aimed at surmounting the few remaining obstacles,
- identify any unforeseen issues, and
- provide an early demonstration that will illustrate the promise of the technology to funding authorities and the public.

This white paper describes the technology, provides some illustrating examples, and identifies currently available enabling technologies as well as technology needs which such a concerted effort could address. It also suggests a near-term demonstration program integrating available technologies as a means of stimulating research and raising awareness of the potential for Smart Spaces. This demonstration would make use of existing infrastructural assets that would minimize the capital cost of such a program.

2. Background

The short history of computing reveals a definite progression in information processing (IP) paradigms. Centralized mainframe computing managed by highly specialized operators gave way in the late 1960s to remote terminals and shared centralized IP. In the mid to late 1970s, the appearance of workstations and networking technology caused another shift to one of distributed IP. By the late 1980s, relatively inexpensive processors were appearing on every desktop, leading to the notion of computers as a personal resource and ultimately bringing about the client-server IP model that is largely in use today. The most recent shift in this progression came with the explosive growth of the Internet in the early 1990's, heralding an unprecedented amount of information sharing in a completely dispersed, heterogeneous "cyberspace".

Looking forward to the next decade and beyond, a continued decline in the hardware price/performance ratio will cause processing resources to be incorporated in more and

more commonplace appliances that today are “dumb” objects. Cars already have processors to perform control functions, but information management and display will become a standard feature as well. Cellular telephones with rudimentary information management/display capabilities are currently being fielded. Personal digital assistants equipped with wireless communications capabilities are about to enter the market. Building lighting, HVAC controls, and shared facilities (such as conference rooms and briefings chambers) will soon be managed and actuated by processing elements. Internal and external data networks will have dramatically increased processing capability to deal with the explosion in bandwidth and interconnectivity requirements.

This proliferation of processing power will cause another IP paradigm shift: the emergence of dynamic network computing (DNC). Whereas today’s IP paradigm binds software to the hardware on which it executes and regards networks largely as a means of transferring data between processors, the DNC concept envisions a world of interconnected smart appliances acting as a vast resource to be interacted with and exploited by users equipped with portable personal devices. Wired and wireless communications channels will be interwoven to provide seamless connectivity between fixed and mobile assets. Mobile code will provide the means for sending tasks to execute wherever needed in the system, regardless of differences in system configuration, allowing software tasks to move as freely as data to the place where they are most efficiently executed. Load balancing, a consequence of the ability to redefine where processing occurs, will alleviate performance bottlenecks and exploit currently wasted instruction cycles. Information services will become more accessible from remote locations, and the properly equipped user will have enhanced control over his environment. The end effect of this level of information system integration will be to realize the full potential of the information age on the conduct of commercial, military, and even personal human affairs.

3. The Dynamic Network Computing Concept

3.1 Generic Concept Description

The DNC concept assumes an infrastructure providing connectivity among a large array of fixed and mobile processing devices in the environment. Connectivity may be through wireless or wired means. Buildings, for instance, would be equipped with processing nodes in each room that are interconnected with local and wide area nets, but that also provide a wireless means of connecting with mobile processing agents in their immediate area. As they enter and leave various spaces, the mobile agents register their presence in (or departure from) the space with static assets. While in the space, each mobile agent negotiates access to shared resources within the space such as shared displays, communications channels to other agents in the area or the external world, and even computational resources (processing time and/or storage).

Because of fundamental limitations of the mobile agents, they will depend upon the infrastructure to provide IP services they do not have the internal capacity to perform. As these portable devices move among zones, the infrastructure will dynamically reallocate resources to adapt to the changing clientele. Properly implemented, those changes will be largely transparent to the user. As they move among differing environments (such as from an interior environment to an exterior one), the mobile agents will have to adapt in various ways. They may have to adapt to varying communications media and protocols. They may encounter varying levels of infrastructural services. The system (portables and infrastructure alike) must gracefully deal with communications dropouts and subsequent recovery in the midst of performing tasks that are divided between them.

These issues aside, the combination of shirt-pocket computing, mobile code, and infrastructural support will place all the functionality of the desktop environment and

much more at the fingertips of people on the go. Properly equipped users will always be connected, taking advantage of the resources present in their current local environment and adapting as the user moves from one "infosphere" to another. They will be able to access a large information space and manipulate their environment with unprecedented ease. Likewise, properly equipped vehicles will interact with the local infrastructure, providing operators with up-to-date information on the status of the environment, assistance in moving through the space, and ready access to data while remote from his/her home facilities.

Central to DNC is the notion that objects being passed about in this dynamically configured information space will include not only data, but methods as well. Processing tasks will be performed in the infrastructure as a service to a performance-limited mobile clientele. User preference profiles will be managed by dynamic software agents that can be passed around the system to maintain consistent, user-defined style of performing tasks, displaying information, and managing connectivity. Routing will be a combination of domain-based and location-based routing. Since resources are dynamically allocated, resource availability will be a key attribute to be tracked, predicted, negotiated, and managed in a secure fashion. User authentication and registration will become a routine function in most transactions to assure authorized use of services or manage charging for paid services. The network itself will become more transparent and therefore as easy to use as today's telephone network.

3.2 Sample Applications

To illustrate how this paradigm works, we have prepared several vignettes illustrating applications of the technology. Though a host of other applications can easily be envisioned, these serve to highlight the potential as well as the issues of this concept.

3.2.1 Vignette #1: DNC Enables the "Smart Workspace"

The goal of DNC is to provide productivity-enhancing tools to people as they move through their environment and to remove the physical ties to their desktop location, ultimately increasing the effectiveness of people on the go. Consider the businessperson arranging a conference of coworkers, all of whom typically have full schedules. His scheduler queries a smart conference room that indicates available time slots. It then queries the coworkers' scheduling agents to negotiate a time suiting the majority. Once a suitable time is agreed upon, it tells the conference room to allocate the time block. As the meeting convenes, the attendees bring their PDAs or other portable devices containing unique personality modules. These PDAs connect with the conference room infrastructure by wired or wireless means. The presenters automatically download presentation data and configure the display devices in the room according to their preference profile. Conferees can tie into their desktop file structure to pull out needed backup data that can be routed to displays in the room without leaving the room. In the background, the contents of the discussion are captured through speech-to-text capabilities and stored. User-defined filters could later identify a synopsis of key points for those who were absent.

As the various parties adjourn and move through the workspace, they register their location with the infrastructure. Calls, e-mail, scheduling reminders, and documents from colleagues can all be routed to them directly. Since their portable devices have limited capability, the processors in each zone they enter provide background time for such tasks as running documents through various reformatting filters, compressing or encrypting data, and managing data requests from various servers. These fixed assets would likely have a high bandwidth connection to data servers and have no problem transferring large files in their original form, but would strip out unwanted parts compress the data before sending it across bandwidth-limited wireless links to the user's portable device. This

stripping and compression process would be defined and directed by a mobile code agent that the user configured and stored on his desktop or in a central file server to be accessed as needed by the infrastructure.

3.2.2 Vignette #2: DNC Extended to the “Smart City”

A logical extension to the “Smart Workspace” is the “Smart City.” Users enabled with portable devices that are supported by a city-wide infrastructure would have a range of services not possible today that enhance efficiency and effectiveness. Business travelers away from their home city could tie into information kiosks that would locate and provide directions to particular addresses. These kiosks would produce travel advisories that would help avoid congested routes. Continuous connectivity with the home office environment would assure that time-sensitive information would be immediately available while the traveler was in transit. Searches through digital yellow pages would help them locate needed services such as restaurants of a particular type or local document production vendors. In airports, travelers would be able to connect with the airport infrastructure to automatically note their presence at the gate and thereby automatically check in, reducing time spent waiting in line. While waiting for departure, they could tie into the external communication net to perform administrative tasks such as responding to e-mail, processing voice mail, or accessing and editing documents.

Personal business would be equally enhanced by the smart city. For instance, a visit to the local mall would be made more convenient if shoppers who have particular needs identified could tie into the mall infrastructure to locate stores that have a particular item. For those who are browsing, they could survey what stores have sales and specials worth noting. Movement within the mall would be made easier through the use of automated directions and mall maps on their personal display. Transactions could be authorized using secure digital signature methods. Tourists could use the same information kiosks used by business travelers to locate sites of interest. While at those sites, their visit would be made more meaningful by the infrastructure providing information on particular points of interest in a user-interactive manner.

3.2.3 Vignette #3: DNC Enhances SUO Mission Effectiveness

Small units operating at forward locations can also be made more effective by DNC. Consider the job of a covert sensor placement platoon, burdened with not only sensors to emplace but weapons and supplies as well. DNC would provide them with a powerful set of tools in a very small hand-held device that would be far less burdensome than the wearable computers being fielded today. As they traverse the battlespace, soldiers can request maps and imagery to preview areas they intend to move through. The request and response data would be manipulated by the infrastructure to limit bandwidth and provide security. Operations orders, such as sensor placement requests, could be communicated more precisely and securely. Sensor data can be passed to fixed assets that can quickly translate raw data into meaningful information and to be passed to the troops on the scene.

This vision of managing a sensor platoon’s activities requires significant IP capability that is well beyond any portable system. Processes such as sensor data fusion and planning require significant processing power. Imagery and maps require substantial data storage and retrieval capacity. Communications channels are usually bandwidth limited, and if the bandwidth exists, information security policies often restrict data flow. All these issues tend to result in compartmentalizing the information space. For instance, today’s architecture makes the products of sensor data fusion available only to intelligence analysts, but some of that knowledge can be equally valuable to operators in the field as they work to make the most of their sensor assets. Rather than limiting

various functions to specific locations, DNC can facilitate timely movement of information all the way down to individual soldiers on the battlefield.

3.3 Issues Driving The Paradigm Shift

DNC will become a highly desirable if not necessary way of reducing the cost of the information management infrastructure and improving its efficiency. Today's IP paradigm has a number of deficiencies that the explosion of smart devices will severely exacerbate. Driving issues include the following:

Wasted computing capacity — In general, the average workspace today has far more computing capability than is required. This excess capacity is largely due to the fact that idle processors waiting for their human masters are much less expensive than idle people waiting for their processing slaves. As a result, there is an uncountable number of wasted processor cycles lost each day in almost every venue. With the number and variety of embedded processors increasing exponentially, the amount of wasted computing power will increase exponentially as well. At the same time, frustrating delays caused by overburdened server support or saturated networks may continue to be a common occurrence under the current IP paradigm. DNC holds the promise of helping balance the processing load by moving tasks to idle processors whenever possible.

Increased mobility of users — Whether it's in the commercial workplace or on the battlefield, the need to push information down to mobile users is growing. In business, people increasingly need access to their information services (voice and text messages, e-mail, and even file serving) while on the go to maintain their information edge in a competitive environment. In combat, timely planning and situation assessment data has always been an issue, but in the 21st century the winning edge will increasingly be with those who maintain information superiority. That means more and better information arriving where it's needed, with fewer intermediaries and a minimum of latency in the request-acquisition-distribution cycle.

Limited mobile device performance — While the capability of portable devices has grown dramatically, it will continue to be true for the foreseeable future that their capability will be limited compared to fixed assets. Battery limitations, heat dissipation, storage density, communications range, and display capability in hand-held devices will limit how much on-board processing can be done by these devices. A DNC infrastructure that augments the power of these devices when needed can dramatically improve their utility.

Data access restricted by application-specific formats — Currently, format differences restrict data access even when network connectivity exists. For instance, even with state of the art equipment, one cannot readily access the content of a Microsoft Word document on his or her Digital PCS. Data reformatting is the answer, but the code resources for format conversion among all possible types cannot be carried on small portable devices. DNC can mitigate this limitation by providing not only seamless connectivity, but also the ability to recognize and correct format discrepancies in the infrastructure as documents and data move from source to end user.

Bandwidth reduction and security needs — To alleviate the competition for limited communications bandwidth created by increased use of graphics, imagery, and real-time audio, data compression will become more widespread. Dynamic computing allows greater flexibility in permitting a variety of compression formats to be employed and downloaded as needed. The location of compression/decompression tasks can be dynamically determined depending on bandwidth of available channels and processing burden. Similarly, a heightened need for information security will also drive the need for local processing, but variable styles and formats of encryption may overburden mobile

processing capability. DNC can facilitate this process by serving the appropriate methods to mobile components on an as needed basis.

4. Supporting Technologies

4.1 Extant Technology Enablers

In addition to cheap, ubiquitous processing, there are a variety of technology developments that, if properly integrated and exploited, makes the advent of DNC possible:

Java and the birth of mobile code — More than just a language, Java provides a run time environment capable of executing instructions received from a remote source, regardless of whether the sender's and receiver's processing environments are compatible. This has facilitated the emergence of mobile code — code resources that can execute wherever needed. Although Java's portable code capability is used today mostly as a means of dynamically managing client-side user interaction, it clearly has the potential for defining methods that can be executed by remote clients. This means that objects can be communicated containing not only attributes and parameters, but methods as well. At present, security concerns have restricted Java to a limited domain of execution (the so-called "sandbox") that excludes allocating computing resources and accessing files remotely, but research is underway to mitigate these limitations safely.

Personal digital assistants — This year will see the advent of personal digital assistants enabled with wireless communications. For the first time, pocket-sized devices with bit-mapped displays will be capable of accessing and displaying all kinds of information remotely. With DNC, they will also serve as a front end to a computing infrastructure of almost limitless dimension. The downside is that the variety of devices currently entering the marketplace will present a large variety of user interface types that may challenge applications designers.

Digital signature — One of the problems with mobile code is that it opens the door for utilization and/or corruption of computing and data storage resources by third parties. Public key encryption technology provides part of the answer through reliable digital signatures. Providers of code resources and their clients can perform data and code sharing transactions and be assured not only that the data or code has not been corrupted in transit, but also that the parties in the transaction are authenticated.

Markup languages — The success of HTML as a data representation standard has stimulated a variety of other markup languages, including Dynamic HTML (DHTML), XML, and the Handheld Device Markup Language (HDML). HDML is interesting in the present context in that it is designed for devices with limited processing and display capabilities. This standard will enable such devices, most notably Digital PCSs and PDAs, to interact with data servers. In addition, extensions to these languages can easily provide a universal registration grammar for managing resources in a dynamic network environment.

Advanced protocols and media — High performance networking is the goal of numerous development efforts by both the government and industry. Technologies such as Asynchronous Transfer Mode (ATM) and Wave Division Multiplexing (WDM) are also pushing the state of the art toward orders-of-magnitude performance improvements. Low performance wireless media, such as infrared signaling, can also provide connectivity for low cost processing elements within the "Smart Space". However, a network with such disparate bandwidths can have significant performance-limiting bottlenecks. DNC can provide load balancing of bandwidth and processing to alleviate these problems.

RF Tagging and other mobile ID forms — User authentication can be performed as a software function, but hardware-based technologies are also emerging that can reduce or eliminate the processing burden. While hardware solutions may reduce flexibility, they may provide an important efficiency and may meet some special security needs that software-based approaches do not satisfactorily address.

4.2 Technology Need Areas / Suggested Research Directions

In spite of the maturity of many of the technologies, there remain some areas for research and development work. Our preliminary assessment of this field indicates that the following technology areas would significantly accelerate development and implementation of DNC. While they are not barriers to a convincing demonstration of the benefits of a DNC infrastructure, these technologies will ultimately be needed enablers for practical implementation on a wide scale:

Mixed mode location routing and domain routing — Today's networks use domain-based routing. This protocol works well for fixed assets or environments where topology reconfigurations are relatively infrequent. However, dynamic network computing will require a seamless blending of domain-based and location-based routing. The need is analogous to the problem of routing telephone signals to cellular clients, but will be complicated by the need to define smaller and more numerous "information cells". These cells would have to be capable of regulating their physical dimension and restricting access to those entities physically present. Depending on the transmission medium, reconciling overlapping spaces will be essential to the success of the concept. For instance, if low-power RF is the medium of choice, one issue is how devices in adjoining rooms do not interact if they should be logically separated.

Identification of, and auto-registration with resources — To make DNC reality, communications and representation standards will have to be developed so that mobile users can interact with a variety of environments. Registration with a given infrastructural domain and access to its resources will require a common language and a set of common communications media that will work equally well in both confined spaces (such as within an office building) and in less structured spaces (such as on the highway or on an open battlefield). Establishing what infrastructural resources are available and applicable for a given need will require protocols and representations that are consistent throughout the infospace.

Generalized, standardized public key infrastructure — To be able to move freely about and interconnect dynamically with various environments, accessing and exploiting resources dynamically, the architecture must be able to provide user authentication and assure secure access to private or restricted information. While public key encryption technology and the capabilities it affords (digital signature and authentication of parties in a transaction) can answer this need, an infrastructure for managing keys and styles of encryption will have to be developed. Depending on the domain being managed, this infrastructure may be public or private. However, what is clear is that a cost-effective open architecture for such infrastructure will be required for truly seamless connectivity across a heterogeneous set of environments to become a reality.

System and information security — Whether in the business community, in private homes, or on the battlefield, the paradigm of DNC with mobile agents tapping into infrastructural resources adds a new dimension to some of the old security issues. How does one guarantee that only authorized users can access resources? How do you manage access authorization without imposing an undue amount of bureaucracy? How does one prevent unauthorized access or tampering? What "back doors" are created by the use of mobile code, and how and when must it be inhibited? How is a security infrastructure put in place that does not slow the system to a crawl and is virtually transparent to a user

community that needs information security but does not want to be burdened by its use? What level of security is right for various applications or domains?

Standards for sharable code module libraries — If every method to be used in a given process has to be served across the network, the demand for bandwidth could quickly become intractable. Some methods must be resident in local processing resources. However, different configurations will have different resident methods and hardware capabilities. Standards for how locally available methods are identified and communicated to servers will be essential for ensuring that the desired data can be effectively exploited, and for efficiently determining what methods are required to complete a given transaction.

Network computing simulations and performance measuring tools — As the DNC paradigm is explored and implemented, tools for measuring the performance impact of various architectures and approaches will be needed. Tools for predicting performance problems and bottlenecks will be essential in designing workable configurations. These tools will have to be capable of simulating the activities of hundreds and thousands of agents conducting millions of transactions. System stability in the presence of dynamic task reallocation and an unprecedented degree of dynamic resource allocation will be a serious concern that will need to be tested before widespread implementation is attempted.

5. Conclusion

Dynamic networked computing integrates a variety of technologies to provide new efficiencies in how we manage information processing, and ultimately how we conduct human affairs. Some of the enabling technologies are in existence today, including personal digital assistants, data encryption and security services, mobile code, mobile agent ID and authentication, and common data formatting protocols. Others are just over the horizon, such as wireless PDAs, ultra-high bandwidth (Gbps) media, and ubiquitous processing). DNC is well suited for phased, incremental implementation, and it could be demonstrated in a limited form in the near term with existing technology. There are even infrastructure installations available today that would be good settings for such a demonstration.

For this vision to become reality, a few technological issues of modest dimension must be resolved. A coordinated program is needed to focus the effort on the technology need areas and issues identified above. Some of these issues can be resolved by experimental implementations. Others require a disciplined, coordinated design effort. Still others require developing a community consensus on standards that will quickly gain acceptance and withstand the test of time. The most effective way to resolve these issues is to embark upon a coordinated, a scenario-driven demonstration program using application vignettes as an organizing vision (similar to those offered in this paper). Working to a shared vision will energize and focus efforts across a diverse and distributed research and development community.

Appendix: Company Capabilities

1. Corporate Overview

Science Applications International Corporation (SAIC) is one of the largest suppliers of information, data security and network solutions in the world. Clients in both the commercial and government worlds turn to SAIC to develop new ways to use information technology (IT) to improve productivity, increase revenues and reduce costs. In 1997, as a part of its effort to establish preeminence in the information technology marketplace, SAIC acquired Bellcore, one of the world's largest telecommunications engineering and consulting companies and a leading provider of information networking software.

The combined SAIC/Bellcore company offers a breadth of IT and telecommunications experience far surpassing most other firms. SAIC/Bellcore brings its clients full, end-to-end solutions encompassing management consulting, engineering consulting, software, program management, systems integration, outsourcing, education and training, and many other disciplines. Independent surveys consistently rank SAIC among the top information technology companies in the world. Our success derives from strong partnering relationships with clients and in-depth expertise in all facets of the IT field.

Founded in 1969, the company has had over 28 consecutive years of increasing revenues and earnings. With the recent acquisition of Bellcore, the company and its subsidiaries have estimated annual revenues of nearly \$4 billion. Headquartered in San Diego and International in scope, SAIC has offices in over 150 cities worldwide. SAIC exists to deliver best value services and solutions based on innovative applications of science and technology, in serving Federal, Commercial, and International customers.

2. SAIC Information Systems Integration Programs

SAIC has a distinguished track record for large project management and systems engineering in mission-critical, budget-constrained environments. Our portfolio of projects spans a very diverse range of technical applications areas. A few of our recent and ongoing information technology projects include:

- The Composite Health Care System (CHCS) for the U.S. Department of Defense, the world's largest and most advanced patient information system. This system permits the documentation of clinical treatment strategies and provides the opportunity – through the use of a single data base covering over 9,000,000 patients – to compare the outcomes of specific intervention strategies. In Government Executive Magazine's November 1996 issue SAIC's CHCS Project is called "one of the most successful systems integration programs of the decade."
- The NISE program, under which SAIC provides commercial and military Ultra-High Frequency (UHF) Satellite Communications (SATCOM) support for Navy ships, submarines, and shore facilities. SAIC provides design engineering, pre-installation test and checkout, fabrication, installation, software development and maintenance, testing, and lifecycle logistics support. SAIC provides integration services for systems as diverse as slice radio, Joint Internet Controller (JINC), battle force e-mail, below and above decks shipboard communications systems, video teleconferencing systems, and shipboard Internet and data links communications systems.
- The FBI's Integrated Automated Fingerprint Identification System (IAFIS), for which SAIC is designing, developing, and implementing an on-line criminal history data base using a client-server architecture that will support Federal, State, and local law enforcement agencies via the National Crime Information Network. SAIC is

providing hardware, software, software development, system testing, integration, documentation, training, and maintenance.

- Visitors to the Atlanta Olympics were able to get up to the minute information on traffic conditions, accidents, weather, flights, bus schedules, tourist attractions, and general directions from 130 interactive kiosks throughout Georgia. Built as part of the GDOT's Advanced Transportation Management System, the system is designed to improve transportation in the region by providing better traffic monitoring and surveillance and by making that information available to the public. The kiosks have been installed at MARTA rail stations, state visitor information centers and rest areas on interstate highways, shopping centers, hotels, office buildings, and metro Atlanta airports.
- Nearly 3,000 toll lanes in the United States and Pacific Rim countries rely on toll collection equipment integrated and maintained by SAIC.

For more information, point your browser to the URL: <http://www.saic.com>

3. Bellcore Communications Technology

3.1 Mobile Communications:

Bellcore provides a wide range of solutions to both high- and low-tier wireless service providers such as cellular, paging, satellite, and personal communications services (PCS) operators. We understand mobile communications and interconnection needs, and have the necessary core competencies to put Intelligent Network (IN) solutions for PCS and cellular to work in diverse applications.

Mobile providers, local and long distance telecommunications, cable, and other wireless carriers are looking for alternative distribution models. As these models are developed, the need to integrate hardware, network, and software will predominate. This is where Bellcore's technical expertise can help mobile operators become competitive and profitable.

Our verification and testing processes are respected throughout the industry. And because security and fraud issues are extremely critical and sensitive to the mobile industry, Bellcore has applied world-renowned security expertise in communications network security to help clients evaluate, design, and implement existing network security features, remote system security access, disaster planning, database security methodologies, and physical site security features.

3.2 Telecommunications

Bellcore has more than 800 clients in 55 countries, using 130 operations support systems with 85 million lines of code

- The recognized architect of the interoperable U.S. telephone network. Eighty percent of public telecommunications in the U.S. depends on software invented, developed, implemented or maintained by Bellcore.
- Nearly 800 patents for technical innovation. Bellcore developed breakthrough technologies such as AIN (Advanced Intelligent Network), ISDN (Integrated Services Digital Network), and video-on-demand. Bellcore also developed the network systems used to handle every 800 and 888 call placed in the U.S.
- A recognized leader in internetworking existing and emerging technologies and helping clients benefit from broadband, wireless, and Internet technologies.

3.3 Network Instantiation and Management:

At Bellcore, we intimately understand the opportunities and challenges facing network implementations. Our services are unique for their end-to-end perspective, and include comprehensive interoperability testing at strategic points in the process. Equally important is the safety and protection of the network, which includes prevention, containment, and disaster recovery, plus response services for network outages and security intrusions.

Bellcore holds the patents for much of the technology that is driving network communications today (such as AIN and ISDN). In fact, we have designed the network systems that handle all 800 and 888 calls in the United States, and are putting our expertise to work in the international arena for companies seeking to develop world-class networks. Bellcore routinely provides the full range services for entities creating or changing networks, including planning and engineering, operations, interoperability testing, integrity and reliability assessments, security and fraud prevention, and disaster recovery and prevention.

3.4 Regulatory Issues:

Bellcore's regulatory experience and knowledge base spans vast areas of FCC, federal and state regulatory activities, and International communications. Bellcore's unique ability to apply that expertise on an independent and objective basis, through our organization and participation in industry standards seminars and other cooperative committees, is recognized throughout the expanding communications industry.

Bellcore's integrated products and services cover a wide spectrum. We draw from our extensive experience – in the communications and networking industries, as well as in the field of requirements – to design solutions that can help develop the competitive advantage you need. Changes impacting all communications companies include competition in the local exchange business, competition in both intra- and inter-LATA toll, number portability, unbundling and access. The ability to compete in these areas calls for sound planning, the right business processes, and cost effective operations systems.

For more information, point your browser to the URL: <http://www.bellcore.com>

4. Network Solutions

In 1992, Network Solutions won a competitive bid from the National Science Foundation to be the global registrar for the top-level domains (TLDs) of .com, .net, .org, .gov, and .edu. Network Solutions has sustained the stability of the Internet by continuously delivering reliable services. Along the way, we have registered nearly 2 million domain names worldwide.

Network Solution, as a subsidiary of SAIC, has successfully managed the exponential growth of an increasingly commerce-oriented Internet. As a pioneer and leader in Internet technologies, Network Solutions has been helping clients solve networking and internetworking related problems for over 18 years, counting among its clients multinational oil and gas corporations to leading financial institutions.

On March 11, 1997, VeriSign, Inc. and Network Solutions announced the availability of "one-stop registration" for organizations wanting to establish a secure presence on the Internet. For the first time, domain name registrants have the option to enroll for a VeriSign Digital ID as part of the registration process.

Digital IDs are a key component in enabling secure communications through the industry-standard SSL (Secure Sockets Layer) Protocol. The majority of industry web

servers, including servers from Microsoft, Netscape and other leading vendors, are already enabled to support VeriSign Digital IDs. This alliance between Network Solutions and VeriSign makes it easier to obtain a Digital ID and activate a web site's security.

For more information, point your browser to the URL:

<http://www.netsol.com/nsi/index.html>

DEVICE INTERACTION IN SMART SPACES

Bill Mark

SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025
(650) 859-4530 office
(650) 859-6171 fax
bill.mark@sri.com

ABSTRACT

Smart spaces are envisioned as user-aware environments defined by a changing set of interacting devices. Considerable emphasis is being placed on how this collection of devices will interact with users in the space. Here I explore an underlying issue: How do devices interact with each other to achieve a desired result?

1. THE PROBLEM

Smart spaces will be situated collections of devices, where "situated" means that the role of any device will depend not only on its own characteristics, but also on its situation. The situation of a device depends on the other devices in the space at any given time. This set of devices may change, either because devices enter or leave the space, or because devices within the space go in and out of operational readiness. Even the concept of "space" is situated; e.g., it may be appropriate to expend more time or power to interact with a larger set of devices in some situations.

I focus on the following issues:

- *Connection*: how do devices start interacting?
- *Reference*: what do devices call each other?
- *Content*: what do devices say to each other?

2. CONNECTION

As devices come and go, they need to self-connect with other devices. I am not addressing the communication aspects of how devices find each other in the space. Instead I explore the issues of how they begin interaction once they have established contact. Connection will almost certainly start with a lowest common denominator handshake protocol much like fax machines establishing a mutually acceptable communication bandwidth. But in smart spaces there will be heterogeneous devices; interaction requires more than agreement on bandwidth. For example, devices will not know what they can say to another device until they identify its type.

Plug-and-play environments like modern PC's use very low level protocol to establish the identity of new devices. These protocols are based on the assumptions that

1. there are relatively few kinds of devices (interaction software for all the devices the PC will ever encounter are either preloaded or can be downloaded when required); and
2. the environment is trusted (i.e., that no untrusted device will be sneaked on to the port).

These assumptions are not appropriate for smart spaces.

Internet style interaction uses only lowest common denominator protocols, and is too inefficient for smart spaces.

A possible approach (and one that is being

implemented in next generation PC environments) is the use of mobile code exchange to establish interaction information. Once communication has been established via the lowest common denominator start-up protocol, devices exchange class identifiers, which identify them as members of broad (and stable) classes like handheld display devices, cameras, etc. After the device knows the other's class, it can exchange whatever interaction enabling software (e.g., drivers) it has for devices of the that class. The software will be in the form of mobile code, with the usual assumption that the receiving device has a virtual machine that can execute the code. If both devices can successfully incorporate the received code into their own interaction environment, more optimal communication can take place.

Any improvement in interoperability immediately brings up security concerns: self-connecting devices could be a security nightmare without concomitant authentication technology. A secure co-processor could provide the necessary hardware environment, but the authentication scheme is yet to be worked out (Do all devices have a worldwide unique identifier? How does the system keep track of which identifiers are "okay"? Should authentication stop at the device, or extend to the user of that device)?

3. REFERENCE

What do devices call each other? Certainly, all of them can have unique names, but we don't want them to have to know each other's names in order to converse. As the number and diversity of devices grow, we will need to have a reference paradigm that includes non-arbitrary groupings and relationships. For example, the model should support references of the form "John's notepad", "all the notepads in this space", "all the notepads of people in my unit", "notepads that are being used right now", etc.

Some of these references depend on intrinsic characteristics of objects (e.g., people are identified by personal names and have possessions like notepads). Other references depend on extrinsic properties that can change over time (e.g., service readiness; location [for some types of objects]; possession [what is my notepad now can be someone else's notepad later]).

These different bases of reference have important implications for creating an overall reference paradigm. For example, the concept "in this space" is defined *only* geographically, and defines a set that may vary over even very short time periods due to mobility. The "personal name" concept defines a set that changes orders of magnitude more slowly. Many things do not move at all over short periods of time, so that location becomes very like an intrinsic property (e.g., "the wall display in Bldg. E"). Another large class moves only relative to a "container" -- e.g., a building -- that does not move. Containers do not have to be geographical (e.g., "my unit" is a container for people. Finally, virtually any form of intrinsic reference can be combined with extrinsic references ("the notepad of everyone in my unit who's in the space now").

Smart spaces must allow all of these forms of reference, a problem well beyond the scope of Internet Protocol (IP) addressing and the current mobile IP schemes [1, 2, 3]. All the same, smart space reference paradigms must work into IP addressing (or whatever form of internet addressing is chosen in the future). Current IP addressing is based on "containers" -- organizations in this case -- arising history and politics. Since there are not that many containers of this kind, it does not matter very much how you address containers (for most applications). And, since there is a significant legacy investment in the current scheme, it may stay in place for a long time. Smart space addressing will then be a hybrid scheme that uses historical groupings to get to the container, and semantically defined references within that container or among containers, as illustrated above.

For referring to the location of mobile devices within a space, the approach of addressing with respect to landmarks like "the east wall" or "the main console" is probably more useful than a GPS-style location. For referring to mobile devices outside of the space, but within a container, either landmark ("on this floor") or GPS-style would work. For addressing mobile devices outside of the container ("within a 10 mile radius"), a GPS-style scheme is very compelling.

4. CONTENT

Finally, the “what do they say to each other?” problem. After devices make connections and establish some form of reference, there is the very large issue of content exchange. What can a notepad assume about other devices when its owner walks into a space? These devices may be other notepads, cameras, pointing devices, wall displays, etc., each with its own characteristics and functions. The notepad software contains a model, implicit or explicit, of the kind of content it can take as input and use as output. But what about other devices? Which ones can give it a particular kind of content, and how does it ask for it?

I stress “content” because the issue goes beyond data typing. As discussed in the first section of this paper, the interpretation of a request for data depends on the situation of the device. For example, a person in a smart space may circle a vehicle icon on his notepad, gesture toward a wall display and say “this vehicle will be 300 meters north of that building in five minutes”. The speech recognizer, wherever it is located, must interpret this utterance in the context of the vehicle represented by the icon on the notebook, the particular display indicated by the gesture, and the map and building shown on that display. In fact, the smart space may have to determine what the person was pointing at by seeing which display in the general direction of the gesture was showing a building (presumably) in a map context.

This kind of interaction can work only if the various devices can make assumptions about the kind of content that other devices “know” about. The speech recognizer needs to resolve the referent of the phrase “this vehicle” by looking for a vehicle-type object that is salient for the speaker [4]. Something in the smart space must know that salience can be indicated by notebook gestures. There must also be knowledge that the circled icon represents a vehicle. Resolution of “that building” requires similar knowledge. As mentioned above, this resolution process could in turn be used to determine the devices that need to be in the interaction.

A number of different architectures could be used to implement smart space device interaction. All smart spaces could have a central server that “dispatches” requests from all devices. The smart space could be organized in terms of agents and mediators [5], where the mediators have knowledge about which agents (in

this case devices) have which kind of information. Or, each device could broadcast to all other devices or a subset of devices, assuming that “someone out there” will have the right information.

No matter what architecture is chosen, there must be some sort of framework shared by potentially communicating devices that incorporates the basic set of common assumptions or “shared knowledge” they can rely on. These frameworks will be based on the characteristics of the devices in the space and the tasks that will be performed.

Assuming that the number and diversity of devices will grow, and that tasks will change constantly, it becomes imperative for this framework to be based on non-arbitrary models of the world. If every smart space and every task has its own *ad hoc* content framework, device interoperation will be nearly impossible within a space, much less among different spaces.

Non-arbitrary models, or ontologies, are being developed in a number of domains [6], but the approach must be adapted to the specific needs of communicating devices. A first step is to recognize that the actions in the domain are communication actions, in particular interchanges among devices. Then, taking the point of view of individual devices, catalogue the “speech acts” each class of device is capable of (e.g., a notepad can *tell* the contents of its display, *ask* for speech recognition services, etc.) Devices will fall into clear classes based on capability (display, gesture, speech I/O, etc.), allowing significant inheritance and providing structure for the modeling of new devices.

Next, given the speech acts for a particular device, represent the specific information interchange capabilities and requirements among a given set of devices being used for a particular task. What does a notebook need to say to another notebook or to a wall display in a map-based tactical command-and-control task? Again, tasks can be put into a class structure that provides inheritance and organization.

This approach is related to that used in KIF and KQML [7], except that the focus on device interchange in smart spaces provides significant leverage in creating a shared representation of content. KIF is a generic representation language,

intended to cover any interaction. Here the idea is to go beyond generic representation to specific ontologies of the concepts required to represent device speech acts and smart space tasks.

Besides the narrower focus, resolving the representation problem in terms of the separate dimensions of devices and tasks reduces the overall complexity. The different dimensions define largely decoupled class hierarchies; when a specific device is used in a specific task, the content of its interchanges is represented by the combination of the appropriate classes from the different hierarchies.

Finally, an interesting and potentially far-reaching aspect of the content problem involves the dimension of time. Given the large variety of devices and smart spaces, it will become incumbent on devices to keep track of their own history -- and to pass it on to their successors before they "die". This is important for authentication (knowing the pedigree or *provenance* of an artifact is an important authentication technique). But it is even more important for providing stability from the human point of view in a stressfully dynamic environment of different devices and spaces. (Wouldn't it be nice if your notepad had "tribal memory" of how to coordinate with other devices to give an icon salience on a display in particular space. And wouldn't it be nice if your notepad remembered how you finally got electronic cash last time you were in Ulan Bator...)

on the Practical Application of Intelligent Agents and Multi-Agent Technology, (Blackpool, Lancashire, UK), The Practical Application Company Ltd., March 1998.

6. <http://www.teknowledge.com/HPKB/participants.html#techDev>.
7. Tim Finin, Yannis Labrou, and James Mayfield, *KQML as an agent communication language*, invited chapter in Jeff Bradshaw (Ed.), "Software Agents", MIT Press, Cambridge, to appear, (1995).

REFERENCES

1. IETF IP Routing for Wireless/Mobile Hosts (Mobile IP) working group, <http://www.ietf.org/html.charters/mobileip-charter.html>.
2. Mobile IP Overview, http://www.cis.ohio-state.edu/~jain/cis788/mobile_ip/index.html.
3. D. B. Johnson and D.A. Maltz, "Protocols for Adaptive Wireless and Mobile Networking," IEEE Personal Communications Magazine, February 1996, pp. 34-41.
4. A. Kehler, JC Martin, A. Cheyer, L. Julia, J. Hobbs and J. Bear, *On Representing Salience and Reference in Multimodal Human-Computer Interaction*. AAAI Workshop on Representations for Multi-modal Human-Computer Interaction. Madison, Wisconsin, July 26-27, 1998.
5. Cheyer, A., D. Martin and D. Moran, *Building distributed software systems with the open agent architecture*. Proc. of the Third International Conference

Trajectory-based Adaptation

Based on a Submission to DARPA RFI 98-04, in the Topic Area of Smart Spaces¹

Murray S. Mazer and Charles Brooks²

The Open Group Research Institute
11 Cambridge Center
Cambridge MA 02142 USA

1. INNOVATIVE CAPABILITY ENVISIONED

The confluence of wireless technology, mobile computing, and intelligent objects will result in environments within which "smart spaces" can be created, permitting mobile users to interact with intelligent objects in these spaces. We propose a capability by which, as the user moves through one "smart space" toward another, the infrastructure determines which smart spaces are likely to be next (based on the user's "trajectory" through the current smart space) and adapts those spaces to serve the user's needs. Among other benefits, this capability can be expected to provide at least an order of magnitude more responsiveness in applications across smart space boundaries (by substantially reducing latencies associated with handover).

Examples of adaptation include the following:

- As the user moves from one "smart space" to another, the infrastructure arranges to pre-fetch the user's applications and data to the 'next' space(s). The benefit is that this activity reduces latencies associated with network-based applications and data access.
- As a refinement of the previous item, the infrastructure automatically provides high availability of cached data and applications by replicating and/or striping "locally" within the new smart space. The set of participating devices is defined by the set of "intelligent objects" reachable and capable of contributing necessary resources for the anticipated lifetime of the requirement.

- As the user moves from one "smart space" to another, the user's applications adapt to the human interface capabilities in the new space. Applications change opportunistically to use the "best" human interface capabilities. Moving to a "better" capability may mean expanding capability offered to user; moving to a "worse" capability may mean shedding functionality.

This approach can provide benefits regardless of whether seamless connectivity is actually achieved. If seamlessness is possible, then the user experiences reduced latency and adaptive interfaces; if seamlessness is not achieved, then the user also experiences higher availability through caching.

2. POSSIBLE BENEFITS AND APPLICATIONS

The possible benefits of this approach are detailed in the above section. A program of research activities would include both the determination of trajectory and use of trajectory information for specific adaptation.

3a. MAJOR TECHNICAL OBSTACLES

In order to realize the above potential, several technical obstacles must be overcome. These obstacles exist both in the determination of user trajectory and in the use of that information for different kinds of adaptation (such as the interface, cache, and high availability adaptation listed above). Some example obstacles include:

- Determining the user's trajectory through smart spaces.

¹ This research was supported in part by the Defense Advanced Research Projects Agency (DARPA) under contract number F19628-95-C-0042. The views and conclusion contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Project Agency or the U.S. Government.

² Corresponding author Mazer is now at Curl Corporation, Cambridge MA USA (mazer@curl.com). Brooks is now at GTE Internetworking, Waltham MA USA (clbrooks@bbn.com).

- Identifying, pre-fetching, and caching applications (including mobile code caching).
- Identifying, pre-fetching, and caching data.
- Managing caches (to avoid oscillations or eliminating something from the cache if user will likely return to the area).
- Determining the capabilities of "intelligent objects," in terms of storage, protocol, etc. We assume the existence of "intelligent objects" in the smart spaces with very high amounts of storage and with a high-quality network connection.
- Dealing with smart space security and accountability issues, including authentication of the user and the user's mobile device to smart spaces; authorization for application and information access; authorization, authentication, integrity, and privacy of caches within smart spaces; veracity of trajectory sightings; and allocating/arbitrating resources among competing interests. All of these issues require a notion of who the user is and the rights accorded to her; in addition, a smart space environment will have many more principals (intelligent objects and devices) than is found in current environments, implying scalability requirements.
- Supporting cache discovery: the user's system, when it moves into a new smart space, must discover the caches in the new space (through a handoff mechanism from the old space, or through a discovery protocol).
- Defining and maintaining consistency between one smart space cache and the authoritative sources for the cache, especially if the user is allowed to update elements in the cache.
- Defining and maintaining consistency between one smart space cache and its counterpart in the next smart space, especially if the user is allowed to update elements in the cache.
- Identifying the kinds of data or applications for which trajectory-based adaptation is a reasonable approach.
- Adapting human interface capabilities: for "follow-me" applications that use the smart space's human interface capabilities, the resulting applications may be less efficient (i.e., less customized to particular platform assumptions), but they may also be orders-of-magnitude easier to develop, especially if the infrastructure automatically provides certain support capabilities (e.g., interface teleporting).
- Designing smart-space-adaptive applications: in the presence of trajectory-based smart spaces, application design becomes more complex. Applications must be inherently resilient (through toolkits or in applications themselves) to changes in the application environment, such as in interface capabilities. The traditional "client" component of applications (that which runs in user device) may need to be restructured to accommodate a piece that runs on the human's portable device and partly on a device in the smart space where the user happens to be.
- Translating application interfaces and output. If application interfaces are going to be presented dynamically and adaptively via interface capabilities in the smart space, then there must be ways to define the application interfaces so that they can be exported to the smart space. Similarly, we need to translate application output for display and user input for application control.

3b. PLAUSIBLE APPROACHES

We foresee several plausible approaches for overcoming the obstacles described above.

As an aid in trajectory determination, an implementation may use mobile base stations and the "intelligent objects" in the smart spaces to track the user's system.

In order to identify applications for pre-fetch and caching, a system may utilize "traditional" means for identifying such applications, plus smart-space-specific specifications. The same observation holds true for data identification for pre-fetch and caching.

Applications and data for use in trajectory-based smart spaces may be based in mobile code, which delivers itself (prepared to continue) to the next smart space. Intelligent objects, within a given space, may have protocol capability downloaded to them (e.g., to make an object capable of participating in a replication protocol).

Consistency issues regarding shared data and state are generally understood; however, such issues will need adaptation-specific evaluation. Resource allocation and arbitration may be achieved by using a cost-based scheme. Finally, cache priming in a new smart space may occur by transferring cache contents from the old space, rather than gathering the cache data from its original sources. This requires a clear definition of cache consistency and assumes some forms of contract and cooperation between smart spaces.

Mobility Management 'Straw' Roadmap

Murray S. Mazer¹ and M. Ranganathan²
Mobility Management Working Group
DARPA/NIST Workshop on Smart Spaces
30-31 July 1998

Introduction

This note attempts to provide some structure for the discussions of the Mobility Management Working Group at the DARPA/NIST Workshop on Smart Spaces. The note first focuses on the types of mobility that might be encountered in the envisioned Smart Spaces world. Then, based on the mobility types, we identify three axes for discussion: applications, technical challenges and opportunities, and existing technologies.

Types of Mobility

The charge to the Mobility Management Working Group states that mobility management concerns "the ability to locate, stage, and present relevant information to users as they move through Smart Spaces, which can include rooms, buildings, vehicles, and individuals." The focus is on ensuring that the user can continue to interact with relevant information in the face of mobility. Although users are the most obvious mobile entities in a Smart Spaces world, there are others, and it serves us well to identify them in order to understand their impacts on mobility management.

We have identified eight kinds of things that might be mobile in a Smart Spaces world:

1. The human user
2. Devices that the user carries (such as laptop, PDA, watch, belt buckle)
3. The user's applications and application context (which may travel in the user's devices or by separate means, including mobile code)

¹ Curl Corporation (mazer@curl.com)

² NIST (mranga@herbivore.ncsl.nist.gov)

4. The user's data (which may travel in the user's devices or by separate means)
5. The user's network context (such as connections to services).
A potential point of controversy is whether seamless connectivity should be assumed for Smart Spaces. Also, seamless connectivity does not imply constant levels of service (bandwidth, etc.)
6. Elements of the user's physical/smart context (for example, the user may pop out her favorite light switch at home and take it with her on a trip, inserting it into a provided spot in her hotel room).
7. Infrastructural properties (e.g., one Smart Space may offer a certain capability to the user which is not necessarily available in the user's next Space—the new Space may need to import that capability)
8. The Smart Space relative to the surrounding Space. For example, a vehicle providing a user's Smart Space may move down a highway toward a node of high bandwidth at which the Smart Space may download some missing functionality that it has arranged to acquire at the node. The user may be in control of the Smart Space's movement (e.g., while driving an automobile) or subject to the movement as controlled by someone else (e.g., while flying in an airplane).

It may be instructive to consider the various combinations of the above types of mobile entities, rather than each type in isolation or all types at once. This is because the deployment of Smart Spaces will likely be phased in over time and will not accommodate all types of mobility in a single epochal event. A related goal is to identify the timeframes within which various "mobilities" will be achieved and the obstacles to achieving them.

Axes for Discussion

Potential Applications and "Big Ideas"

One of the goals for the workshop is to identify applications that could be enabled by Smart Spaces. The application area deeply influences the infrastructural requirements for mobility management. For example, the key mobility management issues in a battlefield scenario are likely to be significantly different than those in an office scenario. In identifying applications, it is worthwhile to state explicitly which of the types of mobility is assumed and what constraints are present (for example, security constraints).

Another goal is to identify some new "big ideas" (such as *Active Information*) that could be used to make Smart Spaces a reality.

Areas of Technical Challenge and Opportunity

The “mobilities” listed above affect a variety of areas of technology. In order to prepare a research agenda to help us move toward supporting mobility management in Smart Spaces, we should identify (at least in list form) the affected areas. We can further elaborate on the challenges (i.e., why current technology is deficient) in each area to the extent we understand them. The areas include (in no intentional order):³

- Networks
 - Intelligent (active) networks
 - Overlay networks
- Physical devices
- Application structuring
- Application programming interfaces and toolkits
 - Should mobility be exposed to, or hidden from, applications?
 - A uniform interface definition for the interaction between the application and the (distributed) mobility management infrastructure.
- End-system operating systems
- Smart Space infrastructure (including inter-Space capabilities)
 - Techniques for composing individual services into larger-grained services
 - Techniques for importing capabilities into a Smart Space
- Human-computer interfaces
 - Techniques for exposing and discovering device interfaces
 - Techniques for presenting information in user-relevant ways within the capabilities of the available interfaces

³ The second level bullets are meant to be suggestive of issues that we might identify within each area of technical challenge.

- Techniques for composing interactive interfaces to individual and composed services
- Collaborative environments for mobile users
 - Interaction modes for users of smart spaces
 - Building mobility awareness into collaborative environments.
- Mobile code
 - Security and cache management
- Caching and replication of user-specific, application-specific and system-specific information
 - Techniques for identifying and staging relevant information
- Middleware and component technology (such as JavaBeans, ActiveX, and Tcl extensions)
 - Techniques for exposing and discovering service interfaces
 - Adding mobility to component technologies
- Location sensing and tracking
 - Architecture of location management infrastructure
 - Granularity of tracking
 - Replication of tracking information
 - Interaction of location-tracking infra-structure with devices
 - Types of location queries supported
- General infrastructural issues for mobility management
 - Security
 - Scalability of the Smart Space infrastructure
 - Fault-tolerance
 - Interfaces between the various technical areas
- Interoperability, policy, and commercial issues

- Interfaces between Service Providers (e.g., if WorldCom operates one Smart Space and Yahoo operates another, they must support the required mobilities between their spaces.

Existing Technologies and Gaps

Some Smart Space Mobility Management requirements may be supported by existing technologies (either directly or, more likely, through extension), whereas other requirements may demand fundamentally new approaches. Identifying the relevant existing technologies (and the concomitant gaps) helps us to clarify areas in which fundamental invention must occur.

Areas in which a considerable amount of research has dealt with relevant issues of mobility include: Active Badges, Mobile Phone (PCS) networks, Mobile Object Systems and Mobile agents, Intelligent agents, Mobile IP, Proxy support, and Caching. What are others? What aspects will likely prove relevant to Smart Space Mobility Management, and in what ways do current capabilities fall short?

The Potential for Military Use of Augmented Reality Technology

David Mizell, Boeing

“Augmented Reality” (AR) is a technology which consists of three main hardware components:

- a wearable, usually belt-mounted, PC
- a see-through head-mounted computer display (HMD)
- a 6 degree of freedom head position and orientation tracker.

An AR system can be used to superimpose computer graphics or text on specific coordinates of a real-world object. Because the computer is receiving very frequent updates of the user's head position and orientation, the graphics in the see-through display can be made by the computer to appear to the user to be stabilized on particular positions on the object, almost as if they were painted there. AR is directly related to fighter aircraft head-up displays (HUDs) and the helmet-mounted gunsights used by military helicopter pilots, which use similar computations to accomplish similar goals with see-through display optics. The distinguishing characteristic of AR systems is that they are designed to be used in a hands-free mode by people who may be moving around within a fairly large work area.

At Boeing, we have been developing, prototyping and experimenting with Augmented Reality systems since 1990. From 1994 through 1997, we were funded by a DARPA TRP project to prototype and test AR systems and simpler wearable computer systems, running real applications, under realistic conditions. These applications included industrial manufacturing tasks, and commercial and military maintenance tasks. For the AR system, the primary application task we have studied has been wire bundle assembly in aircraft manufacturing. The AR system guides the worker through the routing and sleeving of a wire bundle via computer-generated diagrams which appear to the worker to be drawn on the formboard on which the bundle is being assembled.

Virtual Reality and Augmented Reality technologies, with their immersive or see-through head-mounted displays, respectively, have a great deal of potential for 3D information visualization in the military context. The problem is that the military must be mobile, and currently available position/orientation tracking systems are not. They nearly all require house current, some are built into ceilings, and nearly all are designed

under an assumption of a fairly permanent and carefully-surveyed placement of some of their components.

These current attributes are not inherent to the technology, however. It is possible to have small, battery-operated head trackers. Bootstrapping algorithms are possible, which could determine the exact location of tracking beacons using the tracker system itself, as an initialization step. It is possible to develop a 6DOF head tracking system that would be battery-operated, easily portable, and easy to set up in a new location.

If a 6DOF head tracker is portable, self-contained and easy to set up and use, then the question of military applications becomes a question of when is it of value in the military to have interactive, spatially-stabilized, 3D information visualization? Many significant application opportunities exist:

- *Command/control* - It would undoubtedly be of value for commanders and their staffs to see a 3D visualization of the battlefield. Right now, this capability is effectively restricted to situations where a fixed infrastructure can be expected: the CIC of a ship, or a specially-equipped van, for example. A portable, easily-deployable head tracker would enable use of this technology in any command post or bunker in which ground force commanders chose to set it up.
- *Maintenance and maintenance training* - Maintenance and maintenance training are expensive and problematical issues for the DOD. Military weapons systems are steadily becoming more complex, and many of their components are becoming more reliable. Both trends decrease the probability that, when something does break, someone is around who is qualified to fix it and remembers how to fix it. It is easy to imagine an augmented reality system being used to guide a less-qualified person through a maintenance procedure, pointing out each part that has to be removed in its turn. The problem is that there does not exist a 6DOF head tracker that can easily and quickly be attached to the aircraft, ground vehicle or other weapons system that needs the maintenance.

In the maintenance training context, we should note that AR would not be very effective at teaching a person how to back out a Phillips-head screw. Its strength would be in guiding a user through the correct steps of a procedure. However, this is one of the most significant issues in maintenance training. In both

commercial and military aircraft maintenance, there is a high incidence rate of "No Fault Found." The maintainer, thinking he remembers the correct fault isolation procedure, fails to consult the manual. This causes him to take erroneous steps in the troubleshooting procedure, resulting in a correctly-functioning component being pulled off the aircraft.

Much of the enabling hardware technology for AR applied to these military applications is commercially available. Wearable PCs are available off the shelf from several vendors. HMDs with see-through optics are similarly available. The one enabling technology that is not currently available off the shelf is a suitable 6DOF head tracker. The ideal head tracker for military use of AR would be

- highly portable
- easy to deploy, calibrate and register into the coordinate system of the workpiece or workspace
- lightweight and low power, suitable for being body-worn -- untethered
- accurate to .01", .1 degree or better
- fast, providing measurements at 50 Hz or better
- impervious to acoustic or EM interference
- reliable and repeatable.

Boeing's AR wire bundle assembly pilot project in the Everett, WA factory during the summer of 1997 used a videometric tracker developed by TriSen, Inc. of Minneapolis, MN. It was fairly fast, accurate and repeatable, but fails for general military use on the issue of easy deployability: it requires that the workpiece be painted with an array of fiducial "polka dots" which are detected by the tracking system.

To understand the significance and the absolute necessity of this deployability requirement, consider an example scenario of a soldier using an Augmented Reality system to guide him through the repair of a tank engine. Let us assume that the soldier has basic automobile repair skills but no experience with this particular engine. First, he puts on the wearable computer belt and the see-through AR HMD. Next, he clips the tracker beacons onto designated positions on or around the engine. Then he performs a simple calibration procedure. Now the AR system and its user are registered into the coordinate system of the tank engine, and the AR system leads him through the steps of the maintenance procedure one by one, showing him each part to remove and providing verbal or visual instructions on how to do that removal – the ultimate realization of "Just In Time Training."

In a similar manner, a huge number of other command/control, maintenance and maintenance training tasks would be amenable to implementation in AR or would be candidates for replacement by Just-In-Time AR training, if an accurate, easily-deployable head tracker were available.

Many technologies are applicable to the problem of tracking head position and orientation in real time. In addition to the commercially-available magnetometer-based head trackers, acoustic, optical, inertial, laser, infrared, mechanical, and hybrid systems have been proposed and sometimes prototyped. We know of four approaches to head tracking which seem to have the potential to meet the requirements listed above, including the critical and difficult requirement of easy deployability:

- an acoustic-inertial hybrid system prototyped by InterSense, Inc. Small acoustic speaker beacons can be mounted on or around the workpiece. They emit an ultrasonic beep when they detect an infrared strobe from the HMD. This gives a time-of-flight measurement from each beacon to each microphone on the HMD. The inertial system keeps things on track between acoustic measurements.
- An optical tracker being prototyped by the University of New Mexico. This system uses a cheap, simple microelectronics device called a "quad cell" to get a directional measurement to each of several infrared beacons mounted on or around the workpiece. Each of these beacons can be a small, stand-alone device powered by a watch battery.
- A system being prototyped by VisiDyne, Inc. which uses infrared laser diodes mounted on or near the workpiece and photosensors on the user's head. The laser diodes are amplitude modulated in the 1GHz frequency range. Their signal is detected by the photosensors, and the phase difference enables an extremely accurate position measurement.
- A system prototyped by PhaseSpace, Inc. which uses video cameras mounted in fixed positions and LED beacons mounted on the user's headgear.

Along with the "hardware" problem of designing the 6DOF head tracker, there are algorithmic problems, as well. There needs to be a method of quickly calibrating the tracking system and, at least in the case of maintenance-related applications, getting the tracker (and the user) registered into the coordinate system of the workpiece. "Bootstrapping" methods would be ideal here, because if they worked reliably, the tracking beacons (or whatever the fixed-position component of the tracking system is) could be placed

wherever most convenient, rather than having to be surveyed into known positions.

A Framework for Intelligent Collaboratories

B. Parvin, G. Cong, J. Taylor, and C. Tay
Information and Computing Sciences Division
Lawrence Berkeley National Laboratory
Berkeley, CA 94720
parvin@george.lbl.gov

Abstract

This paper outlines the motivation, requirements, and architecture of a collaborative framework for distributed virtual microscopy and instrumentation. In this context, the requirements are specified in terms of (1) functionality, (2) scalability, (3) interactivity, and (4) safety and security. To meet these requirements, we introduce three types of services in the architecture: Instrument/Sensor Services (IS), Exchange Services (ES), and Computational Services (CS). These services may reside on any host in a distributed system. The IS provide an abstraction for manipulating different types of instruments; the ES provide common services that are required between different resources; and the CS provide real-time visual routines for scientific image analysis and situation awareness. These services are brought together through CORBA and its enabling services, e.g., Event Services, Time Services, Naming Services, Security Services, etc.

1 Introduction

The current trend in collaborative research is to bring experts and facilities together from geographically dispersed locations. The natural evolution of this type of research is to leverage existing computational toolkits to support novel scientific applications based on capabilities in simulation, inverse problem solving, visualization, real-time control, and steering. This paper describes the requirements for distributed virtual microscopy, the proposed software architecture, our experience in implementing this system to meet the current requirements, and the infrastructure needed for the next generation of on-line facilities. Although our application has been applied to the microscopy

domain, the same concepts should also scale to other domains.

Our testbed includes two unique transmission electron microscopes (TEM) that are operated by the National Center for Electron Microscopy, and an inverted optical microscope with applications ranging from material science to biology. From the user's perspective, we establish the desirable requirements in terms of functionality, interactivity, scalability, safety and security. From the designer's perspective, we abstract these requirements into three categories of services that include Instrument/Sensor Services (IS), Exchange Services (ES), and Computational Services (CS). These services sit on top of CORBA and its enabling services. IS provide a layer of abstraction for controlling any type of microscope or sensor. ES provide a common set of utilities for information management and transaction. CS provide the analytical capabilities that are needed for on-line microscopy. The design also maximizes the use of existing off-the-shelf software components.

A unique feature of CS is to provide closed-loop servo control and steering in a collaborative framework, where steering aims at recovering a model from observed images within the collaborative framework. Model recovery is an inverse problem-solving process that attempts to (a) link a specimen's behavior to external stimulation (steering of experimental parameters) or (b) construct a 3D geometric model of an object through user interaction (steering of computational parameters). In general, model recovery is a computation-intensive algorithmic process requiring extensive support high-performance computing and low-latency network infrastructure. Thus, in the absence of available network bandwidth and quality of service (QoS), automation at a local site must be increased. Another feature of CS is in situation awareness for large laboratory environment. Here, multiple cameras with different focal lengths are used to detect

This work is supported by the Director, Office of Energy Research, Office of Computation and Technology Research, Mathematical, Information, and Computational Sciences Division, and Office of Basic Energy Sciences of the U. S. Department of Energy under contract No. DE-AC03-76SF00098 with the University of California. The LBNL publication number is 42089.

pertinent motion and provide focus of attention for remote collaborators.

Section 2 summarizes the system requirements. Section 3 describes software architecture and ongoing scientific experiments and their corresponding computational needs. Section 4 concludes the paper.

2 Requirements

This section outlines four requirements for distributed virtual microscopy and steering within the collaborative environment: (1) functionality, (2) scalability, (3) interactivity, and (4) safety and security. Requirements for dealing with the quality of images in terms of their size and dynamic range are beyond the scope of this paper.

1. *Functionality:* Unique scientific imaging systems are designed to conduct specific experimental protocols. This view must be made visible to remote collaborators together with all the available control operations, their range, and accessibility in different coordinate systems. In this sense, the user should be able to query what can be controlled remotely as well as what kind of analytical capabilities are available on-line.
2. *Scalability:* The scalability requirement spans four dimensions, including (a) the number of active collaborators, (b) the number of different microscopes at different locations, (c) the number of legacy applications and analysis programs, and (d) the types of workstation platforms through which a user can engage other collaborators.
3. *Interactivity:* The software interface must allow users to steer either static or dynamic experiments. During static experiments, user manipulation produces rapid 2D visual feedback that provides work-flow continuity. A simple example of static experiment is to translate the specimen and provide corresponding visual motion as a result of that shift. During dynamic experiments, the system should automatically compensate for rapid perturbation that cannot be corrected by the remote collaborator due to absence of end-to-end QoS. Hence, automated real-time periodic adjustments must be made to control the state of an experiment.
4. *Safety and Security:* The system must provide an interlocking mechanism for safe operation of the instrument. In addition, the system should provide authentication, privacy, and integrity for data communication on demand.

3 Software Architecture

The software architecture should provide the infrastructure and functionalities to meet the requirements. Our system uses an extensible object oriented framework (class libraries, APIs, and shared services) so that applications can be rapidly assembled, maintained, and reused. These objects may reside on any host and can be listed, queried, and activated in the system. The architecture illustrated in Figure 1 bridges the gaps between different services that may reside at any node in a distributed system.

Our system consists of three service categories that sit on top of Iona's ORB: Instrument Services (IS), Exchange Services (ES), and Computational Services (CS). The IS are vertically integrated to provide scalable access to different instruments at various sites. The scalability is achieved through a unique abstraction of several types of instruments. The ES provide generic processes and applications common to multiple types of instrument services. Examples include common GUI services, image management services, and session management services. The CS provide a means for simulation, image analysis, real-time visual routines, and visualization. These core services are specified with an interface that is language independent. Every aspect of our system is based on CORBA that includes Internet InterORB Protocol (IIOP), and is supported by Netscape Communicator.

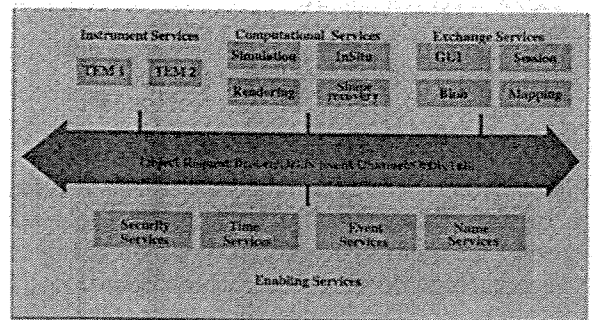


Figure 1: Interaction between various services

3.1 Enabling Services

Data communication between different objects within a server are based on standard CORBA invocation. With respect to interaction between a server and its clients, we have experimented with three models:

- The first one is based on the standard CORBA invocation model of twoway, oneway, and deferred synchronous interaction. Although this model

simplifies distributed processing, it lacks asynchronous message delivery and does not support group communication that can lead to excessive polling.

- The second model uses COS Event Services, which is based on a push-pull model for decoupled communication. An event channel acts as media manager and provides a buffer between consumer and suppliers. A major shortcoming of this model is that clients cannot register with events of interest. In other words, the model does not allow subscription at fine resolution. In general, the Event Services lack parameterized filtering, fault tolerance, access control, and scalability.
- The third model uses OrbixTalk for reliable multicasting, thus, eliminating one of the shortcomings of the previous model. In this model, the clients can "listen" to messages based on their topic and register their interest. Not all routers, however, may have multicasting capabilities.

Our experience indicates that the second model provides an adequate performance for a small number of clients. All messages are time-stamped with Time Services, and delays are periodically relayed back to the server through a secondary event channel to avoid network saturation. The utility of this feature is being investigated at this point.

The clients communicate with servers through the OrbixWeb, which is the JAVA version of Iona's Orbix product. This feature provides scalability on different types of desktops.

3.2 Instrument/Sensor and Exchange services

IS provides a scalable means for instrument control and interaction. For the interaction between classes. The instrument/sensor can then be queried for its properties, which correspond to a control parameter, e.g., focus, magnification. Each operation is validated by the *State* object to ensure a safe transition from one instrument state to the next.

A certain group of parameters in the instrument panel must be periodically read to notify clients of changes. A recent real-time ORB [7] provides such a feature for deterministic scheduling over a high speed network. Furthermore, IIOP has been modified to support end-to-end quality of service (QoS).

3.3 Computational Services

This section presents three examples of computational services that include on-line analysis of scientific im-

ages as well as tools for situation awareness in large laboratory environment.

We have focused on using visual routines to bring new functionalities to scientific instruments[5, 6, 3]. These include insitu electron microscopy and recovery of 3D shape through holographic microscopy (see Figure 2). The former focuses on the behavior of an inclusion (time-dependent morphological changes) as a function of external stimulation, e.g., changes in temperature and pressure. At high magnification, any kind of stress on a specimen manifests with spatial drifts in multiple dimensions. The absence of QoS makes this type of experiment nearly impossible over the wide area network. As a result, computational components are brought close to the experimental setup to analyze the videostream, perform on-line morphological analysis, and compensate for various anomalies so that images at a remote station remain stationary [4]. In this context, steering refers to altering the temperature property on the specimen to study a particular behavior on the inclusion. These on-line measurements from the videostream are performed at 8Hz over the local area network using a symmetric multiprocessing system.

The second type of computational service recovers the 3D shape of an object through holographic electron microscopy, in which a new protocol has been developed [1, 2] to recover the 3D shape of an inclusion from multiple views. Conventional electron microscopy presents projected images with little or no depth information. In contrast, electron holography with coherent illumination provides both magnitude and phase information that can be used to infer object thickness in terms of equal thickness contours (ETCs) from each view of the sample. The holographic images contain interference fringes with spacings (in the best case down to less than an angstrom) in which interference is between the transmitted and diffracted beams. In this context, steering refers to selecting an appropriate set of parameters so that each view of the object can be properly represented for further processing. In addition, a eucentric tilt stage is planned to be brought on-line to improve selection of particular views for reconstruction and hence reduce the complexity of this computation intensive process. Another feature of our system is efficient transmission of 3D data over the wide area network. These 3D structures have been parameterized with hyperquadrics. Thus, each remote client has the ability to render his own view of the object. This is an appropriate representation for inclusions, since they are essentially convex 3D objects.

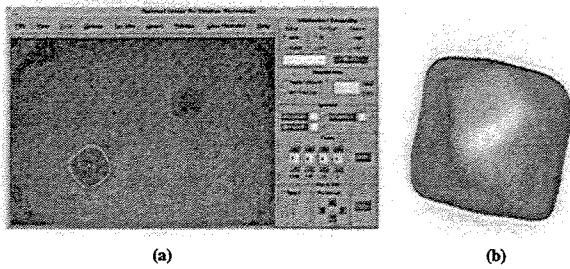


Figure 2: Examples of Computational Services: (a) A specimen is heated during insitu electron microscopy. The red arrow shows the direction of sample drift, but Computational Services compensate for the drift and the position of the sample remains stationary at a remote site. (b) An example of 3D reconstruction from multiple views of an inclusion observed with holographic microscopy.

For situation awareness in large laboratory environment, we have developed a simple motion tracking system with active camera calibration, which detects and tracks moving objects. Motion detection is based on a generic optical flow field algorithm and grouping self similar motion vectors. In general, the flow field cannot handle large spatial motion with continuity that corresponds to real 3D object, and a simple protocol for extraction of motion boundaries is developed for the purpose of initialization. The flow field and subsequent grouping provides a template for correlation-based tracking. These two processes run concurrently to provide continuous updating and correction of template size and its location. The template size is a critical factor as the object distance to the camera center varies during the tracking process. An example of this process is shown in Figure 3. Our implementation is optimized for near real-time performance through a pyramid implementation. The actual processes are threaded for improved concurrency. A unique aspect of our work is in the calibration of intrinsic camera parameters with an active stage and registration of multiple camera views with different focal lengths. Our method is based on finding adequate corners in a natural office environment, moving stage, matching those corners, and constructing the necessary and sufficient equations for recovery of intrinsic parameters. Hence, in the absence of calibration chart, self calibration can be an aspect of the dynamic tracking process.

4 Conclusion

A set of requirements for distributed virtual microscopy was defined together with an architecture and its implementation. Two unique applications of



Figure 3: An example of motion detection, tracking, and active compensation.

this system were reviewed, and a live demonstration will be included in the presentation. We also plan to integrate the Orbix SSL security component to our system. This service provides authentication, privacy, and integrity for data communication over TCP. This technology does not, however, interoperate with multicasting. In fact, the security of multicasting is an active area of research at this point.

References

- [1] G. Cong and B. Parvin. Shape from equal thickness contours. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 1998.
- [2] G. Cong and B. Parvin. Shape from interference patterns. In *Proceedings of the International Conference on Pattern Recognition*, 1998.
- [3] B. Parvin and et. al. Telepresence for in-situ microscopy. In *IEEE Int. Conference on Multimedia Systems and Computers*, Japan, 1996.
- [4] B. Parvin and et. al. Visual servoing for on-line facilities. *IEEE Computer Magazine*, 1997.
- [5] B. Parvin, C. Peng, W. Johnston, and M. Maestre. Tracking of tubular molecules for scientific applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:800-805, 1995.
- [6] B. Parvin, S. Viswanatha, and U. Dahmen. Tracking of convex objects. In *Int. Symp. on Computer Vision*, 1995.
- [7] D. Schmidt, D. Levin, and S. Mungee. The design of the tao real-time object request broker. *Computer Communications*, 21, 1998.

TRACKING MULTIPLE, SIMULTANEOUS TALKERS

D. Dwayne Paschall, Ph.D.

Department of Communication Disorders
Texas Tech University Health Sciences Center
Lubbock, Texas 79409-2073

ABSTRACT

In acoustic environments with multiple, simultaneous talkers, normal-hearing human listeners are able to identify and understand the spoken message of a single target talker in the midst of the other competing talkers. Hearing-impaired human listeners (and computer models of speech recognition), however, often show significantly reduced identification accuracy when the speech of a target talker is embedded in a background of other competing talkers. This paper describes a system for tracking multiple, simultaneous talkers for the purpose of voice selection and segregation. If a speech signal is embedded in a competing noise background that comprises gaussian noise or other noise types with known spectra, compensations may be made that reduce the deleterious effects of the interfering noise. However, if the interfering noise is contributed by another human talker, the spectral and temporal properties of the target signal and noise signal are very similar. The result is that linear filtering techniques cannot distinguish between the signal and noise and statistical estimations of the noise characteristics would also remove components of the target signal. One effective method for improving speech identification accuracy by man and machine is to separate the competing speech signals into the individual speech streams. Once the target voice is segregated from the competing background, identification accuracy is improved for both man and machine. Alternatively, an economy of transmission bandwidth is accomplished using this approach since multiple talkers are potentially able to be combined for use on the same transmission line without interfering with one another. A computational approach based on investigations and models of human auditory processing of competing speech signals is presented.

1. IDENTIFYING THE SPEECH OF MULTIPLE TALKERS

Human listeners with normal hearing are quite capable of correctly identifying the speech sounds of a single talker when embedded in a background of gaussian noise or other forms of spectrally-shaped noise. However, when the competing noise is in the form of speech sounds contributed by other concurrent talkers, the situation is more difficult. In this instance of multiple, simultaneous talkers, the target speech sounds are spectrally and temporally similar to the interfering noise. Thus, simple filtering methods for removing the noise become ineffective. When the listener does not have normal hearing sensitivity, they begin to report perceptual difficulties beyond what would be explained by

a reduction in sensitivity alone when trying to understand what one person is saying to them in these types of multi-talker environments. In fact, hearing-impaired listeners often report perceptual difficulties in multi-talker environments, even before they notice a significantly reduced sensitivity to soft sounds.

Similarly, automatic speech recognition systems exhibit degraded identification performance in the presence of multiple competing talkers. Even when the recognition model has been trained using "noisy" speech signals, the model has essentially learned to make classifications based on reduced peak-to-valley distances in the spectra of the training stimuli. When one speech sound is masked by another, the presence and location of spectral peak information provided by the speech formants of the target voice may be obscured by the competing voice or may be changed as two formant peaks merge to form a new spectral peak with a different center frequency. The result is that spectral peak information used to classify speech sounds is not reliable when any given spectral prominence may be contributed by the target talker or by the other interfering talkers. The problem, then, is to be able to attribute spectral information to the correct talker or to partial out the energy of each talker so that a correct classification may be made.

Models of human auditory processing of concurrent speech sounds [1,2,3,4] suggest human listeners are able to segregate competing voiced speech sounds based at least partially on periodicity information in the acoustic signal. Other investigations [5] suggest that the pitches evoked by each voice in a multi-talker environment contribute to a listener's ability to identify the sounds spoken by the individual talkers. Thus, one approach to improving the speech understanding abilities of both man and machine in multi-talker environments is to model the auditory processes of normal-hearing human listeners in similar situations. The goal is to correctly partition the combined acoustic energy of multiple talkers based on periodicity information so that identification accuracy for the target voice may be improved.

This paper describes a method for tracking multiple, competing voices. The approach would allow "attention" to be focused on one target voice for the purpose of voice segregation from the competing background. The resulting segregated speech signal could then be analyzed for recognition (by machine) or understanding (by humans).

2. THE CONTRIBUTIONS OF FUNDAMENTAL FREQUENCY (F_0)

There are multiple configurations (e.g., the number of concurrent talkers) and settings (e.g., spatially separated or overlapping) in which talkers and listeners compete and interact. The task for the listener is to attend to a single talker and to understand the message spoken by the appropriate person or machine. In simple settings with one talker, this task is relatively easy. In multi-talker environments, however, the task becomes more difficult. Many listeners, in fact, report that they have difficulties attending to a single target talker in the presence of other interfering talkers, even if they have no difficulties *detecting* the presence of the speech sounds produced by the target talker. For machine models of speech recognition, similar difficulties exist. Acceptable recognition performance may be observed in optimal "listening" situations. Performance begins to deteriorate, however, when the acoustic background is contaminated with other concurrent talkers. Even more difficult is the situation when the task is to pay attention to the spoken message of multiple talkers.

Previous investigators have found that when two vowel sounds occur simultaneously to the same ear, listeners are able to identify both vowels easier when there is a difference in F_0 between the competing sounds compared to when the F_0 s are the same [1,7]. Recently, [5] demonstrated a strong correlation between listener's abilities to identify the individual vowels in a simultaneous vowel pair and listener's abilities to rank the pitches of the two vowels. Thus, being able to determine periodicity information about individual speech sounds in a multi-talker environment appears to facilitate identification of the speech signals that are present.

2.1 Single Voice Tracking

Traditionally, speech recognition has been conducted with a single talker speaking into a single, dedicated microphone. Each talker had their own input device and transmission channel. Accuracy of the recognition system could be judged in terms of recognition accuracy for the single voice with little or no interference from competing talkers. In this environment, tracking the acoustic signal of the target talker is not difficult. If a single voice is detected, what did it say?

2.2 Two Simultaneous Voices

In the interactive environment comprising multiple talkers, the recognition task become more complex than the single-talker environment. Here, the voices of two or more talkers may compete for the "attention" of the recognition system (whether human or machine). The difficult task of assigning the correct speech sounds to the appropriate producer of those sounds is created. It now becomes advantageous to identify one or more properties of each talker's voice by which the combined signal may be separated into the individual constituent voices for the purpose of improving the recognition of a target voice and/or parallel recognition of multiple, simultaneous voices.

Competing Talkers with Non-overlapping F_0 s - In some environments of competing voices, the F_0 of each talker's voice may occupy a unique frequency range. Thus, it may be possible to group together the energy of each voice based on the fundamental frequency of the harmonic complex that comprises the target voice(s). Unfortunately, this is not always the situation one encounters.

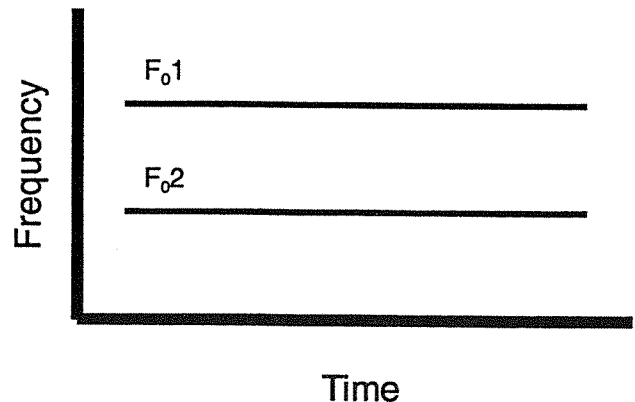


Figure 1. When the F_0 s of two talkers do not overlap, assigning periodicity energy to the correct talker is straight forward because there is never any overlap of F_0 information.

Figure one shows a schematic representation of two hypothetical talkers. The F_0 of each talker occupies a specific frequency region that does not overlap with the frequency region of the other competing talker. In this case, it is straight-forward to group together the harmonics of each talker: simply calculate the frequency of the F_0 in the specific frequency range of the target talker. Then, the energy from voiced-speech segments of the target talker can be reconstructed based on the F_0 information.

Competing Talkers with Overlapping F_0 s - When two or more people talk, it is often the case that at some point the F_0 of the competing voices will either cross in frequency, or come close together in frequency and then diverge [6]. Since several investigations [1,2,4,5] have shown that listeners use F_0 information either explicitly or as a basis for identifying the speech sounds of competing voices, listeners may be able to track or follow individual voices through these periods of crossing and approximation to help segregate the speech sounds of the target voice from the other competing voices.

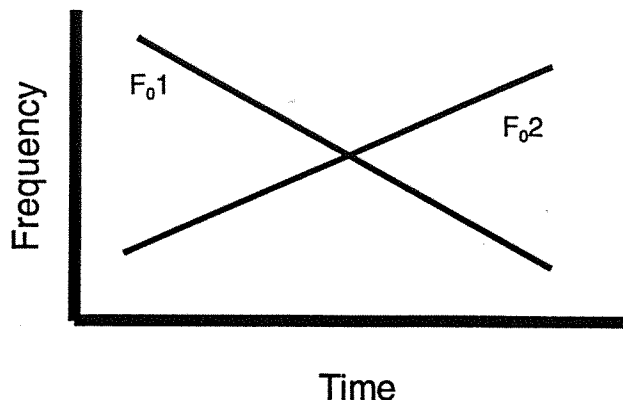


Figure 2. When the F_0 's of two talkers overlap, assigning spectral energy to the correct voice on the basis of F_0 information becomes more difficult.

When the F_0 's of two simultaneous voices are changing, they sometimes converge on each other in the frequency domain where both F_0 's come close together in frequency. When the voice F_0 's then move farther apart in frequency, the listener must determine if the two F_0 's have crossed or simply diverged. Figure two shows a schematic representation of the condition where the F_0 's of two concurrent voices come close together and then diverge. Figure three shows a schematic representation of two voices who's F_0 's have crossed. The task for the listeners is to follow the target (or both) voice(s) through the crossing intersections so that the correct periodicity information is available to them to help the listener identify the speech sounds of the talker.

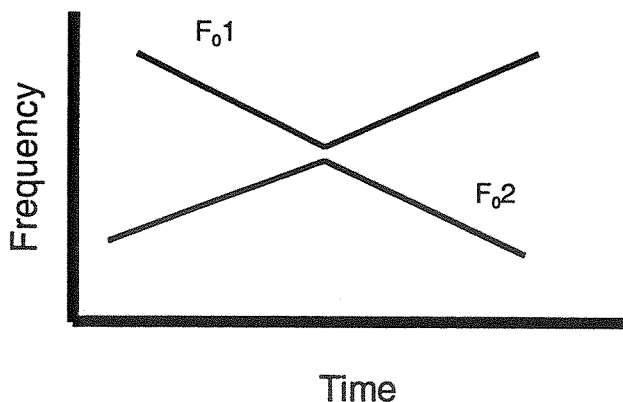


Figure 3. Sometimes, the F_0 trajectories of two competing voices approach each other and then diverge. In this situation, one must determine if the two trajectories have crossed or diverged.

3. TRACKING SINGLE AND MULTIPLE VOICES

This section presents the essential components of a linear neural network approach to tracking multiple, simultaneous human voices for the purpose of voice selection and segregation from the mixture of competing voices. The basic outline of the approach is to estimate the F_0 of the voice(s) using a time-domain subtractive process. The estimate F_0 information is then fed to a linear predictive neural network that predicts future F_0 estimates based on the values of the 3 most recent estimates. This approach works successfully in tracking competing voices, even through periods of crossing F_0 trajectories.

3.1 Pitch Estimation

There are a number of methods for estimating the fundamental frequency of voiced speech. In the current framework, the average magnitude difference function (AMDF) is utilized. The AMDF is computed for a 50-ms Kaiser windowed segment of the speech waveform. This window slides along through the duration of the speech waveform in 10-ms hops. The fundamental frequency of the voice is then estimated as the minimum value in the AMDF. Since the AMDF is a process of subtraction, recurring periodicities produce minima in the function. The minimum with the largest magnitude represents the F_0 of the voice. In the case of two or more competing voices, several minima will occur. In this instance, the peak-picking routine that searches for the minimum with the largest absolute magnitude has a restricted search range. This search range restriction reduces octave and sub-octave estimation errors. In the present implementation, the search range is 80-200 Hz for males talkers and 100-350 Hz for female talkers.

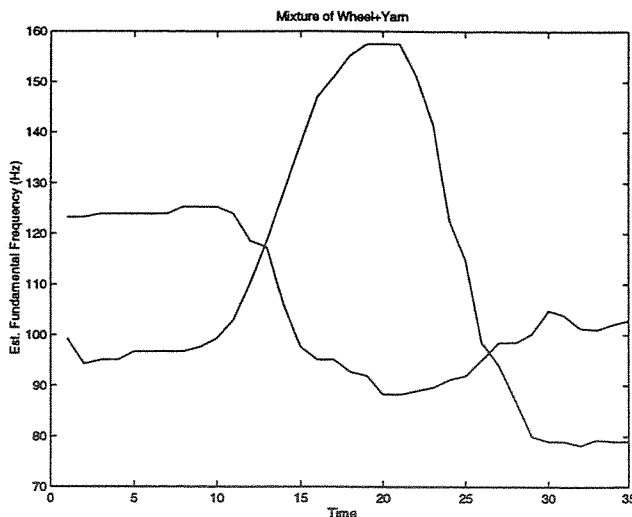


Figure 4. The estimated F_0 trajectories for a males speaker producing the words "wheel" and "yarn" using the AMDF "sliding" window analysis.

3.2 Linear Network Tracking Program

Once the F_0 estimates have been computed, mixture of F_0 trajectories is fed to a linear system model with one neuron model for each trajectory. The inputs to the model are the current F_0 estimate and the two most recent estimates. The next F_0 estimate in time is then predicted from the current and past values. Figure five shows the predicted F_0 trajectories for two words ("wheel" and "yarn") spoken by the same male talker using the linear neural model. As can be seen, the output of each linear neuron predicts the future F_0 values in each trajectory with good accuracy. The error of the network output is less than $2 * 10^{-14}$.

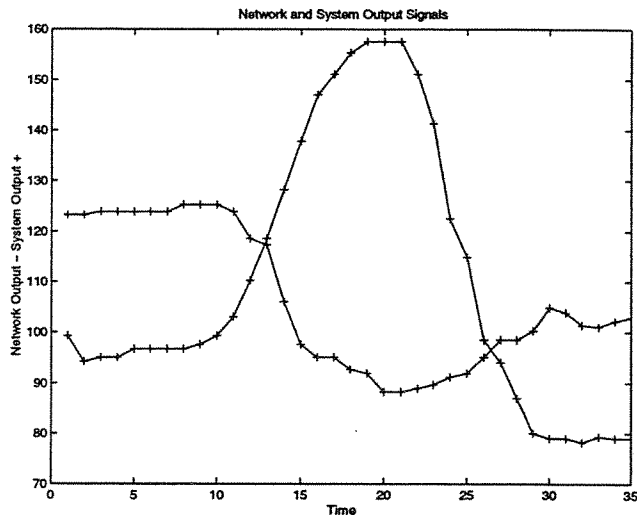


Figure 5. The F_0 trajectories and network tracking of the two words spoken by the same male talker.

4. FUTURE DEVELOPMENTS

To successfully implement a tracking system similar to that outlined above, several related developments may be needed to fully realize a working system. Specifically, there must be some method for determining the number of talkers in a space so that each voice is tracked correctly. In addition, periodicity information is not available for the brief durations of the voiceless speech sounds. Although these time durations are brief, they often contribute important information about the specific phonemic identity of the speech sound. Lastly, recent investigations [8] have demonstrated that the signal processing mechanisms in the normal auditory system are capable of improving the identification accuracy of a target voice by as much as 30% under some circumstances. These biological speech enhancement conditions should be investigated further as a possible method for improving machine understanding of specific voices in multi-talker environments.

4.1 Handling Voiceless Speech Segments

The tracking methods outlined above make explicit use of periodicity information of a particular talker's voice. However, for brief periods of voiceless speech production, these methods may not provide sufficient information for extracting voiceless consonant information. One possible solution to this difficulty would be to adjust analysis window size of the tracking algorithm to average over voiceless phonemes. Then the tracking algorithm would track the F_0 information through the voiceless speech segments. However, a different approach would be needed for longer pauses other than voiceless phonemes (e.g., breathing, stop talking, etc.). The simultaneous voices that were periodic in their speech productions could be segregated from the mixture. This leaves the possibility of using other timbre-related properties for segregating the voices remaining in the mixture. From a psychophysical viewpoint, this seems to be similar to what human listeners may be doing [6].

4.2 Speech Enhancement

When two vowels are presented simultaneously, listeners can identify both vowels more accurately when there is a difference in F_0 between the competing vowels compared to the case where their F_0 s are the same. Figure six shows data from [9] (open circles) that shows that as the F_0 difference between the two competing vowels increases, so does listener's identification accuracy.

HIGH-PERFORMANCE TELE-IMMERSIVE ACTIVE SPACES

Daniel A. Reed
reed@cs.uiuc.edu

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

Michael A. McRobbie
mcrobbie@ovpit.uics.indiana.edu

Indiana University
Bloomington, IN 47405

Thomas A. DeFanti
tom@eecs.uic.edu

Electronic Visualization Laboratory
University of Illinois at Chicago
Chicago, Illinois 60607

Rick Stevens
stevens@mcs.anl.gov

Mathematics and Computer Science
Argonne National Laboratory
Argonne, IL 60439

Maxine Brown
maxine@eecs.uic.edu

Electronic Visualization Laboratory
University of Illinois at Chicago
Chicago, Illinois 60607

Michael Zyda
zyda@cs.nps.navy.mil

Department of Computer Science
Naval Postgraduate School
Monterrey, CA 93943

ABSTRACT

Tele-immersion is an enabling technology for active spaces, unifying *ad hoc* collections of current devices to create coherent, sharable, expandable spaces for information rich environments. However, current tele-immersive spaces are difficult to extend and are sensitive to small variations in network latency or bandwidth. Realizing ubiquitous active spaces will require design and prototyping of new active space hardware and functions; creation of flexible software interfaces for device composition; identification and optimization of network Quality of Service (QoS); and rigorous assessment of human factors, software component interactions, and network dependencies.

1. BACKGROUND

The term *tele-immersion* first entered the research lexicon in October 1996¹ as a succinct description of networked virtual environments, distributed sensors, and digital audio/video in support of advanced computational simulation and intelligent data mining. The pressing need for tele-immersion is a consequent of increasingly distributed, collaborative, planning, research and product development. Today, management, research and engineering teams consist of multi-institutional, multidisciplinary groups who must glean insights from large, distributed data archives, manage remote instruments, and steer complex simulations.

The promise of tele-immersion is collaboration that is more efficient by enabling multiple users to interact in shared virtual or simulated environments, manipulating complex data as if they

¹ A tele-immersion workshop, held at Chicago's Morton Arboretum, was organized by University of Illinois at Chicago and sponsored by Advanced Network & Services. Discussions prompted subsequent meetings, and led to Advanced's National Tele-Immersion Initiative (NTII) [WebRef].

were in the same room [Smith, Foster96] even when they are in remote locales with only portable devices. Implicit in such a promise is low-cost, anywhere/anytime collaborative access.

Tele-immersion is the enabling technology for active spaces, which replace *ad hoc* assemblages of custom, highly expensive components with information-enriched, highly scalable versions of standard environments (i.e., "smart" homes, offices, factories, laboratories, classrooms, and automobiles). By unobtrusively augmenting physical spaces with low-cost immersive displays, environment and device-specific sensors, body and object trackers, intelligent instrument interfaces, streaming audio and video, and haptic manipulators, active spaces, sometimes called ubiquitous computing [Weiser93], add intelligence and data manipulation capabilities to common objects and environments in ways that encourage object composition and extension.

The vision of active spaces is an intelligent unification of current devices (i.e., pagers, personal digital assistants, telephones, fax machines, video conferencing systems, and computers) with emerging technologies to create coherent, sharable, expandable spaces for information-rich environments. These active spaces will consist of a seamless, interconnected web of sensors (primitive and complex), communications devices (mobile and static), distributed, scalable computational resources, variable-latency communication (bits to terabits), large data repositories linked by intelligent agents, and tele-immersion technologies that allow individuals to directly manipulate virtual objects [Stevens96].

The qualitative changes engendered by these active spaces are potentially as profound as those wrought by the telephone and video. Cooperating intelligent objects can unobtrusively suggest and remind, anticipate actions, activate devices, correlate data, and synthesize responses. As part of the workaday defense world, active objects will become the intuitive, comfortable interface for manipulating an increasingly complex world of abstract information and distributed coordination.

2. APPLICATIONS AND BENEFIT

We believe active spaces will catalyze creation of a new generation of intelligent data visualization and manipulation tools in a variety of contexts. Defense and disaster response, multidisciplinary scientific research, and multi-national business all share the need for rapid correlation and analysis of large volumes of geographically distributed data. Active spaces will allow government and defense staff, scientists, business leaders, and government officials to more accurately and efficiently identify emerging trends in heterogeneous data.

As an example, consider a group of defense analysts preparing a report on weather, troop, and materiel movements in the Middle East. Each occupies a tele-cubicle—a workspace with touch sensitive, color plasma panel displays on the walls, work surfaces for conferencing and visualization, haptic devices for direct manipulation of demographic, geographic, climatological and cartographic data, intelligent PDAs that negotiate meetings, and transparent coupling to data archives and high-performance simulation systems.

Within this space, a group of planners might use voice commands to retrieve and overlay demographic and weather data, connect to existing instrumentation in the field, or place read-outs that mine simulation data. They could focus on the Middle East by gaze and relocate US forces in a coupled battlefield simulation by grasping, all while discussing possible logistics and battlefield scenarios.

3. TECHNICAL CHALLENGES

Tele-immersive active spaces presume low-cost, anywhere/anytime collaborative access. However, current tele-immersion systems are both rare and usable only in carefully controlled environments. Even successful and widely used collaborative virtual environments like the CAVE and CAVERN, its tele-immersive variant, are extraordinarily expensive, special purpose systems [Cruz, Leigh97a, Leigh97b]. Assembled from brittle software components, current tele-immersive spaces are difficult to extend and are sensitive to small variations in network latency or bandwidth. For tele-immersion to become a standard mode of collaboration, all these attributes must change.

Realizing ubiquitous active spaces will require design and prototyping of new active space hardware and functions; creation of flexible software interfaces for device composition; identification and optimization of network Quality of Service (QoS); and rigorous assessment of human factors, software component interactions, and network dependencies.

3.1 Active Space Devices and Functions

Virtual environments currently provide primitive direct manipulation of objects presented in proper perspective and facilitate navigation of complex spatial data. Meaningful networked collaboration in these environments is in its infancy. Active spaces add potentially thousands of smart devices that must intercommunicate automatically and under human direction, far exceeding both the real-time capacity of today's operating systems and collaboration software, and the dynamic context movement currently available.

3.2 Software Scalability and Persistence

Today, there is no standard software substrate for adding new devices or functionality to tele-immersive systems, nor can they scale beyond a few distributed sites. Moreover, tele-immersive sessions often retain no state from previous activity. The technical challenges are multifold:

- (a) the determination of a network software architecture that optimizes the available bandwidth and processor cycles,
- (b) a classification of the types of information flow in virtual environments,
- (c) the design of application layer network protocols appropriate to support that information flow,
- (d) the generalization of area of interest management for virtual environments, and
- (e) providing for persistence in virtual environments.

3.3 Networking and Quality of Service

Networking today's virtual environments devolves to low-level socket programming based on UDP broadcast or simple multicasting. Although this is acceptable for small environments, as the number of participants and objects in the virtual world increases, as the requirement for nationwide and worldwide use becomes critical, and as the requirement for heterogeneous interoperability increases, the deep technical challenge becomes designing an effective model for connecting virtual environments with QoS fully enabled.

3.4 Analysis and Usability

To make active spaces truly effective, one must first place tele-immersion on a solid experimental footing by understanding how voice, audio, video, haptics, stereopsis, and other modalities resonate, and by understanding which subset is most appropriate to a particular task, particularly when the computation rate, bandwidth, or latency imposes limits. Moreover, this understanding must guide creation of composable interfaces for active space components that can share information about user behavior and dynamically adapt to changing resource availability.

Meeting these challenges will require quantitative and qualitative assessment of user behavior and creation of real-time tele-immersion measurement tools for dynamic adaptation.

4. TECHNICAL APPROACHES

Current tele-immersion systems target a small domain of computationally intense problems with largely custom, handcrafted software. Active spaces will be low cost, widely distributed, and embedded in everyday objects, with software and interfaces for interoperability. Just as supercomputing has functioned as a *time tunnel*, allowing developers to explore computation models that later become part of the workstation

markets, tele-immersion can play the same role for active spaces. By prototyping active space hardware and software in tele-immersion environments, one can understand the effects of design alternatives, identify common interaction modes, and test software and human interfaces.

This approach would first extend current tele-immersion technologies and systems to form prototype active spaces. The resulting systems will permit accurate human factor and performance studies in relevant application domains. The multidisciplinary group of academic researchers, government laboratory staff, and industrial partners will then develop, deploy, and integrate active spaces with offices, laboratories, vehicles, and homes. This approach relies on the tight coupling of relevant applications, active space devices and software, networking and QoS research, and quantitative analysis and usability.

4.1 Relevant Applications

Experience has shown that research progress is best achieved with one or more driving applications that focus attention on technical problems in a well-defined domain with large potential rewards for their solution [Lehner]. This coupling of computing research with problem domains engages both computer scientists and application specialists in an on-going dialogue about both the potential and limitations of successive active space prototypes.

For example, if an office or laboratory active space is simulated within current tele-immersive environments like the CAVE, one can conduct early design evaluations before complete prototype construction. This model of active space prototyping relies on analysis of test subject behavior when using virtual prototypes of active spaces (e.g., tele-cubicles with intelligent conferencing) [Disz97b].

4.2 Active Devices and Software

The active space of the future combines the best computer graphics, audio, computer simulation, and imaging with fast networking and the ability to track gaze, gesture, facial expression, and body position [DeFanti96]. A sequence of active space reference implementations, each tracking evolutionary improvement in active hardware and software, are needed to conduct performance evaluations [WebRef, CAVERNUS]. Key devices include the following:

Smart tags that attach to all objects, enabling active space components to locate and query objects. Key challenges include low power management and transmission, sensitive accelerometers, and tracking technology.

Data slates with smart tags that provide physical interfaces to virtual information spaces. In a variety of shapes and sizes, data slates, like magic lenses, exploit user comfort with familiar objects and incorporate multi-modal tactile interfaces for bi-directional data manipulation.

Active audio that provides dynamic scoping via small headsets and radio links. Dynamic scoping activates only the audio components relevant to the current context, allowing users to move among collaborative groups based on current interest and

focus.

Structured spatial displays that dynamically shift content to the relevant context. As users shift to and from mobile use with handheld or head-mounted displays to office or conference environments, data and imagery migrate automatically as well.

We believe one can first prototype active space devices in extant virtual environments, inexpensively testing user interfaces before physical device construction.² Given such implementations, application and computer scientists must then work with colleagues in neuroscience and psychology to assess human emotional, physical, and cognitive states by analyzing and calibrating body, hand/arm, facial motion/gestures and utterances [Pavlovic97b].

4.3 Networking and Quality of Service

Tele-immersion applications have emerged as high-end drivers for the QoS efforts envisioned by the Next Generation Internet and Internet2 [WebRef] leadership – real-time support for audio and video streaming are critical to achieving the feel of telepresence. Moreover, collaboration and haptics are very lag-sensitive; the speed of light itself is a limiting factor over transcontinental and transoceanic distances. Researchers must work with networking vendors to develop new QoS protocols for managing multiple priority (control, audio, tracking, haptic) and non-priority flows (text, video, database, simulation, and rendering).

4.4 Quantitative Evaluation

Controlled, experimental measurements of user behavior and the computational and communication costs of interaction are key to understanding the psychometrics of immersive collaboration. By understanding how users exploit tele-immersion and active spaces to collaborate and how their behavior changes when distributed presence degrades, we can derive rules for composing active space components.

To quantify interactions and associated costs, researchers must develop a lightweight instrumentation infrastructure and set of analysis tools to measure tele-immersive and active spaces [Reed95, Reed94]. Finally, participants in collaborative groups often are members of several projects and have multiple interaction loci, usually with unequal foci. Hence, one should explore techniques for smoothly changing interaction modalities, subject to user preferences and available resources.

4.5 Scalability and Persistence

With rare exception, extant versions of collaborative virtual environments are non-scalable, continually broadcasting state changes to all active sites. Large-scale tele-immersion and ubiquitous active spaces require new models for information

² Two of our industrial partners, General Motors and Caterpillar, currently use CAVEs to design and test vehicle interiors. [Lehner, VRatUIC]

sharing that minimize or eliminate unnecessary network access, thus reserving limited bandwidth (particularly for low-power wireless devices) for necessary transmissions.

Experience with large-scale distributed interactive simulations (DIS) [Zyda97b] involving thousands of interacting entities suggests that Area of Interest Management (AOIM) techniques and protocols for active spaces will be key to scalability, persistence state maintenance, and software composability.

4.6 Software Architecture and Composability

Drawing on experiences developing the successful CAVE Research Network (CAVERN) and the Naval Postgraduate School (NPS) distributed interactive simulation (NPSNET) infrastructures, we believe active space software must be extensible and composable [WebRef]. Such an extensible infrastructure might integrate the CAVERN and NPSNET toolkits, support tele-immersive module compositions with soft real-time guarantees, dynamically adapt to available resources, and integrate audio, video, and other data sources.

REFERENCES

- [Cruz] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE," *Computer Graphics (Proceedings of SIGGRAPH '93)*, ACM SIGGRAPH, August 1993, pp. 135-142.
- [DeFanti96] T. DeFanti, D. Sandin and M. Brown, "The Coming Defenestration: Immersive Environments Without Windows," *IEEE Multimedia*, Winter 1996, pp. 6-9.
- [Disz97b] T. L. Disz, M. E. Papka, and R. Stevens, "UbiWorld: An Environment Integrating Virtual Reality, Supercomputing, and Design," *Proceedings of 6th Heterogeneous Computing Workshop (HCW '97)*, IEEE Computer Society Press, April 1997.
- [Foster96] I. Foster, M. E. Papka, and R. Stevens, "Tools for Distributed Collaborative Environments: A Research Agenda," *Proceedings of the Fifth IEEE International Symposium for High Performance Distributed Computing (HPDC-5)*, IEEE Computer Society Press, August 1996, pp. 23-29.
- [Lehner] V. D. Lehner and T. A. DeFanti, "Distributed Virtual Reality: Supporting Remote Collaboration in Vehicle Design," *IEEE Computer Graphics & Applications*, March/April 1997, pp. 13-17.
- [Leigh97a] J. Leigh, A. Johnson, and T. DeFanti, "CAVERN: Distributed Architecture for Supporting Scalable Persistence and Interoperability in Collaborative Virtual Environments," *Virtual Reality: Research, Development and Applications*, Vol. 2, No. 2, December 1997, pp. 217-237.
- [Leigh97b] J. Leigh, A. E. Johnson, T. A. DeFanti, "Issues in the Design of a Flexible Distributed Architecture for Supporting Persistence and Interoperability in Collaborative Virtual Environments," *SC'97 Proceedings*, Sponsored by ACM SIGARCH and IEEE Computer Society, November 15-21, 1997, CD ROM.
- [Pavlovic97b] V. Pavlovic, R. Sharma, and T. S. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, July 1997, pp. 677-695.
- [Reed95] D. A. Reed, K. A. Shields, L. F. Tavera, W. H. Scullin, and C. L. Elford, "Virtual Reality and Parallel Systems Performance Analysis," *IEEE Computer*, November 1995, pp. 57-67.
- [Reed94] D. A. Reed, "Experimental Performance Analysis of Parallel Systems: Techniques and Open Problems," *Proceedings of the 7th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, May 1994, pp. 25-51.
- [Smith] J. E. Smith and F. W. Weingarten (eds.), *Research Challenges for the Next Generation Internet*, *Computing Research Association*, 1997, p. 20.
- [Stevens96] R. Stevens, "Motivations and Thought Experiments for Distributed Wide Area Heterogeneous Computing," *Proceedings of the Tenth International Parallel Processing Symposium*, April 1996.
- [Weiser93] M. Weiser, "Some Computer Science Issues in Ubiquitous Computing," *Communications of the ACM*, Vol. 36, No. 3, July 1993.
- [VRatUIC] "Virtual Reality at the University of Illinois 5/97," (videotape), Electronic Visualization Laboratory, University of Illinois at Chicago.
- [Zyda97b] M. Zyda, and J. Sheehan, Jerry (eds.), *Modeling and Simulation: Linking Entertainment and Defense*, *National Academy Press*, September 1997, ISBN 0-309-05842-2, 181 pages, <http://www.nap.edu/readingroom/books/modeling>
- [Zyda97a] M. Zyda, D. Brutzman, R. Darken, R. McGhee, J. Falby, E. Bachmann, K. Watsen, B. Kavanagh, and R. Storms, "NPSNET—Large-Scale Virtual Environment Technology Testbed," *Proceedings of the International Conference on Artificial Reality and Tele-Existence*, Tokyo, Japan, December 3-5, 1997, pp. 18-26.

[WebRef] Web Addresses

Advanced Network & Services

www.advanced.org

CAVERNsoft www.evl.uic.edu/spiff/ti

CAVERNUS www.ncsa.uiuc.edu/VR/cavernus

EVL www.evl.uic.edu

Internet2 www.internet2.edu

National Tele-Immersion Initiative
www.advanced.org/teleimmersion.html

Next Generation Internet

www.ngi.gov

NICE www.ice.eecs.uic.edu/~nice

NPSNET www.npsnet.nps.navy.mil

Pablo www-pablo.cs.uiuc.edu/

Pyramid Systems www.pyramidsystems.com

STAR TAP www.startap.net
Tele-Immersion www.evl.uic.edu/spiff/ti

The New EasyLiving Project at Microsoft Research

Steve Shafer, John Krumm, Barry Brumitt, Brian Meyers, Mary Czerwinski, Daniel Robbins

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052



Abstract

EasyLiving is a new project in intelligent environments at Microsoft Research. We are working to make computing more accessible and more pervasive than today's desktop computer. More specifically, our goal is to develop a prototype architecture and technologies for building intelligent environments that facilitate the unencumbered interaction of people with other people, with computers, and with devices. This paper describes our goals, design decisions, and applications of EasyLiving.

1. What Is Next in PCs?

Software developers have had a long time to exploit the capabilities of the PC. While new applications for stand-alone PCs are still coming, it is primarily new connected devices that generate new applications and new markets. For instance, inexpensive color printers spawned desktop publishing for the consumer. Digital cameras are creating consumer demand for photo editing software. The World Wide Web (essentially a way of connecting other computers to your own) is giving us unprecedented access to information and a new reason to own a computer.

We are looking at the physical home and work environments as the next things to connect to a PC. Not only will this encourage new applications, it may allow more natural interaction with computers, reducing the barriers of inconvenience that prevent computers from being used for more everyday tasks.

2. Goals of EasyLiving

EasyLiving is a new project at Microsoft Research with its genesis in the Vision Technology Group. Our goal is to develop a prototype architecture and technologies for building intelligent environments that facilitate the unencumbered interaction of people with other people, with computers, and with devices. We are concentrating on applications where we can make computers easier to use for more tasks than the traditional desktop computer. We envision a home or office of the future in which computing is as natural as lighting. It maintains an awareness of its occupants

through computer vision, responds to voice and gesture commands, knows its own geometry and capabilities, and can be easily extended. The technology we are developing will, for instance, enable a home's resident to make a phone call by simply speaking his intentions from anywhere that he happens to be. The home will keep track of children and pets automatically. It will allow a user to move from room to room while still maintaining an interactive session with the computer, with the user interface migrating along.

Being new, most of our work to date has been in conceptualizing and planning. This paper describes our goals, plans, and early milestones for EasyLiving. In order to make this pervasive yet unobtrusive style of computing successful, our intelligent environment must have three characteristics that we detail in this section: self-awareness, casual access, and extensibility. Given the luxury (and burden) of building a new intelligent environment from scratch, we are faced with many concrete design decisions such as the system's user interface and software architecture. We present our conclusions on some of these issues in Section 2. Section 3 describes some applications of our system, including the first demonstration that we recently completed

2.1 Self-Aware Spaces

EasyLiving spaces must be aware of their own activity and contents to allow appropriate responses to the movement of people and their requests. Such a "self-aware space" knows its own geometry, the people within it, their actions and preferences, and the resources available to satisfy their requests. Some examples illustrate the importance of self-awareness:

- As people walk around in the space, they will move through the fields of view of the rooms' video cameras. The system must know the 3D regions covered by the cameras to know which room a given person is in and from which other cameras she may soon be visible. This will allow, for instance, ringing the telephone only in the room where the intended callee is located and migrating a user interface along as the user moves from room to room.

- The system should be aware of the identity of the occupants. The system could sense “absolute” identity, *e.g.* “This is George Jetson”, or it could, more simply, maintain “relative” identity, *e.g.* “This is the same person I just saw from camera 12.” Such knowledge will enable EasyLiving to apply personal preferences for known occupants such as a contact list for telephone calls. It can also be used to block access to certain devices and data.
- EasyLiving must know what hardware and software resources are available to it and how to use them. If a user interface is to move, the system must know how to present it in the user’s new location, *i.e.* whether the new location supports audio, speech, pointing, or visual display. It must understand which devices are already in use and whether each device is working or not.

More than just populating a space with intelligent devices, EasyLiving will maintain knowledge of and employ combinations of devices and software to satisfy the users’ needs.

2.2 Casual Access to Computing

Users should not be required to go to a special place (*i.e.* the desktop) to interact with the computer. Nor should they be required to wear special devices or markers to have the computer know where they are. EasyLiving’s goal of “casual access to computing” means that the computer will always be available anywhere in an EasyLiving space. Through cameras and microphones, the user will always be able to signal the computer. Since the computer will keep track of users and their contexts, the computer will always be able to signal the users in an appropriate way, and it will know how to avoid being obtrusive. For example, a user watching television could be notified via a superimposed window on the screen, while a sleeping user might not be notified at all, unless the message is important. Information access will be similarly versatile, with the system being able to present, say, an address book entry with whatever output device is available at the user’s location in response to whatever input device is available at the user’s location.

Combined with self-awareness, the goal of casual access leads to a migrating user interface. When a user moves, the user interface of the application can move with him. This would be useful for carrying on a phone conversation as a user moved throughout the home, or it could be leveraged to move an interactive session to a device with higher fidelity.

2.3 Extensibility

EasyLiving capabilities should grow automatically as more hardware is added. Extending the concept of “plug and play”, new devices should be intelligently and automatically integrated. One aspect of extensibility is

the view that new devices become new resources that the system can use at will. If, for instance, a CRT is added to the kitchen, it becomes a new way of presenting information in that space, and EasyLiving will automatically take advantage of it. This is an example of extensibility in terms of resources. Another aspect is extensibility in terms of physical space. If a new camera is added, it not only extends the system’s resources as a new device, it also extends the system’s physical coverage. This means that the system must be able to compute the position and orientation of the camera based only on what it sees through the new camera and any others that share its field of view.

3. Design Issues

Our goals for EasyLiving drive our design. This section discusses some of our particular design decisions for component technologies (sensing & modeling, user interface) and broader issues (software architecture and privacy).

3.1 Sensing and Modeling

EasyLiving spaces must respond to users’ actions and words. While there are many types of single use sensors that can monitor people in a room, *e.g.* IR motion sensors and electromagnetic field sensors, video cameras are the most versatile, longest range, and best understood sensing modality for this task. Cameras give rich data that can be used for tracking and identifying people and objects and for measuring them in 3D. In the context of intelligent environments, cameras have been used to track people in Michael Coen’s Intelligent Room at MIT’s AI Lab[1] and to understand gestures in Mark Lucente’s Visualization Space at IBM Research[2].

To accommodate the video sensing demands of EasyLiving, we are building a “vision module” that gives both color and range images. It will consist of between two and four cameras, packaged together, with control and processing done on one PC. We will use the color image to make color histograms, which have been shown to work well for identifying objects[3]. Our own experiments show that color histograms are effective at re-identifying people that have already been seen as long as their clothing doesn’t change. The range images will come from passive stereo, and they will be used primarily for image segmentation. We plan to deploy several such vision modules in each room of an EasyLiving space. They will be used to detect motion, identify people, sense gestures, and model the 3D environment. Modeling is important so each vision module knows what parts of the room it can see and its location with respect to the other vision modules. Given this information, a person-tracker can anticipate which camera(s) will give the best view of a moving person.

We will also deploy microphone arrays in EasyLiving spaces. These will be able to "steer" toward people talking using signal processing algorithms.

3.2 User Interface

EasyLiving will use traditional, desktop user interfaces where appropriate, and also more advanced user interfaces where possible. In general, the user will be able to choose his or her own interface mode, constrained only by the devices available in the room. For instance, a user might start an interaction using a wireless keyboard and mouse in the family room and then move to the kitchen, continuing the same interaction but with voice and gestures instead. This goal spawns two research issues: migrating user interfaces and multimodal user interfaces.

The migrating user interface, such as the family-room-to-kitchen example above, requires that an application, or at least its user interface, be able to move smoothly from one room to another. In their work on the Obliq distributed scripting language, Bharat and Cardelli[4] used a software architecture that moves whole applications between computers using agents. The application, including its user interface, is packaged as an agent, and each computer contains software that can receive such an agent and start it running. As implemented, this scheme cannot account for changes in user interface modality beyond a change in screen size.

In our initial demonstration, we achieved a simple migrating user interface using Microsoft Terminal Server, which allows Windows NT 4.0 to host multiple clients with windows appearing on networked PC's. In the end, however, we want the user interfaces of EasyLiving applications to change with the desires of the user and the available devices. This will require that the applications be written with an abstracted user interface, which is very different from the style of GUI programming today. However, we also see EasyLiving as a new way to run current applications, with an intermediate software layer that could, for instance, reinterpret pointing gestures as mouse movements and spoken commands as menu choices.

The other major user interface question asks: If users could use more natural ways of interacting with the computer, say audio and video, how would they do it? Audio and video output to the user is well-understood, while the use of microphones and cameras as input devices is not fully mature. At the extreme, EasyLiving could carefully monitor all the actions and speech of each user, intelligently interpreting what they mean. We don't expect to achieve this, and instead we will require the user to specifically address the computer for most interaction. (One exception is that the system will passively monitor the room with both cameras and microphones to know when it should be alert to possible user commands, avoiding the "push-to-talk" problem. It

would then suspend any background activity like periodic geometric modeling to pay more attention to the user(s).)

Used apart, microphone and camera input to programs has been the attention of much research. Speech understanding is available commercially, and many computer vision researchers are working on the tracking and interpretation of human movement such as gestures. The use of speech and gestures simultaneously, however, is a relatively new area of research. Sharon Oviatt has studied the use of speech and gesture in a pen-based, geographic map application[5]. When given the choice to issue commands with a pen and/or voice, users preferred to convey locatives (*i.e.* points, lines, and areas) with the pen, while they preferred speech for describing objects and giving commands. For intelligent environments, this means that perhaps the most important gesture to detect is pointing, while other commands should, initially at least, be left for voice.

We have completed the first in a series of user studies to explore speech and gesture input for EasyLiving. Six subjects were led through a series of exercises using a limited, paper and pencil prototype scenario in which they placed a video conference call from a large display screen on the wall. Given a choice, users preferred speech over gestures, but they could effectively combine both. We collected a broad sampling of the kinds of gestures and speech commands that users generated spontaneously for this task. We have also run paper and pencil walkthroughs of potential 3D user interface designs that could be used for the more advanced user interface directions this project will take. Users provided useful feedback in terms of what metaphors they found to be most meaningful for the "Contact Anyone Anywhere" scenario (Section 4.2) we were exploring. Our goal is to next test the redesigned prototype in a series of "Wizard of Oz" studies using a large wall display.

3.3 Architecture for Extensibility

As described above, one of EasyLiving's goals is automatic extensibility. This appears to be a feature that has not been addressed in other intelligent environment research. Our system will automatically incorporate new devices as they are added. The architecture is shown schematically in Figure 1. At the beginning of an EasyLiving installation, the only "live" device will be a central server. This server will contain all the software necessary for running an entire EasyLiving system. It may, in fact, be remotely located and/or remotely maintained and updated. The central server will also maintain information that is global to the whole system, such as the current time and a directory of people.

Each room of the EasyLiving installation will have its own process called a "room server". When the room

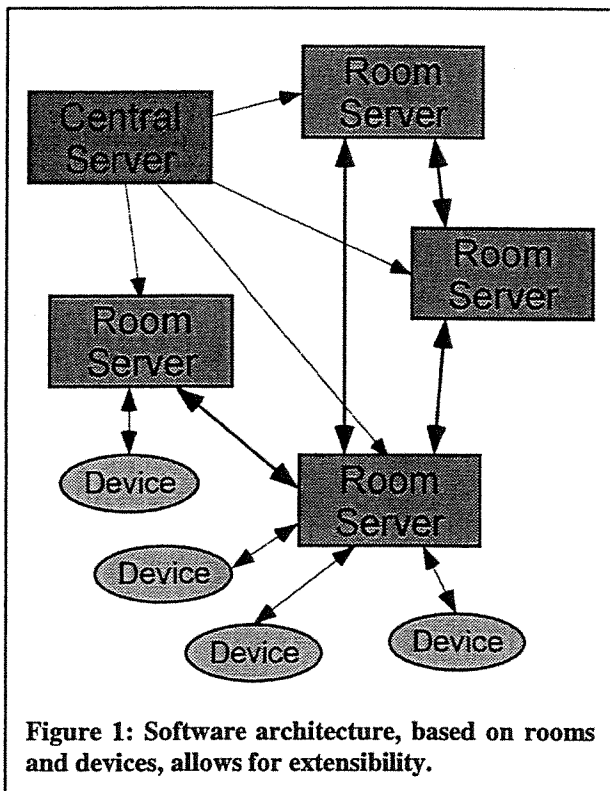


Figure 1: Software architecture, based on rooms and devices, allows for extensibility.

server is activated, it will announce itself as such to the central server and download the required software to make it a room server. This includes all the software necessary for the other processes that may run in the room. As other rooms are added to EasyLiving, they will also have room servers that start up in the same way. Each device added to a room, for instance a vision module, will connect to its room's room server and download the necessary software.

The room server will contain a model of the room, including its geometry, its contents, and locations of people. It will be connected to the room servers of adjacent rooms. These connections will be used to exchange information about overlapping fields of view of the rooms' cameras and to alert adjacent rooms that someone is about to enter.

This architecture will simplify the addition of new devices and new spaces to an EasyLiving system.

3.4 Privacy

In any intelligent environment, there is a tradeoff between privacy and convenience. The more the system knows about you, the more it can do for you, but the more it may reveal to someone else. Just having cameras and microphones in the room begs the question of "Who might be watching and listening right now?" There will also be symbolic data in the system that can be used to infer the users' habits and preferences. These concerns are made worse by the fact that the system

could be passively gathering the data even while the it is not being actively used.

One way to make the system more secure with respect to outside snooping is encryption. We expect that as e-commerce becomes more common, it will provide publicly trusted encryption methods for transmitting data over networks. We may also want to enforce a policy of not transmitting any video over the EasyLiving network, choosing instead to do all computer vision at the camera and only transmitting results. If the user desires, the system can be set up such that it will not try to identify anyone unless they actively request it by using a password, cardkey, or biometrics. This means the system will not know who is in the space.

In general, privacy must be deeply rooted in the system with the tradeoffs made clear to users. There is not a single good answer to the question of making the system actually private and convincing users of the same.

4. Applications

We have completed our first demonstration of the EasyLiving system, and we have several more applications planned. We realize that we cannot predict what will be the most useful and popular applications for EasyLiving. We are confident, though, that the combination of capabilities that we provide will spark new applications.

4.1 Migrating Windows

Our first demonstration (July 1998) showed an implementation of a migrating user interface. We set up an office with three video cameras monitoring three "hotspots" – 3D regions of interest in the room where a user could go to interact with one of three video displays in the room. The locations of the hotspots were drawn as rectangles in the three camera views prior to running, as shown in Figure 2. To be considered in a hotspot, a user had to appear in the hotspot in at least two camera views. Each user began by logging into the system and starting an application at one of the displays. Based on this login, we knew which hotspot the user was in, so the system stored the color histograms of the corresponding image regions. Identifying users with their color histograms meant that the system could accommodate more than one user in the scene simultaneously. The system continuously monitored each hotspot, and when the histograms matched the stored histograms, the application window was moved to that display. The application's window was moved using Microsoft Terminal Server. Each camera had its own, dedicated PC, and a fourth PC ran the terminal server. The PC's communicated via sockets.

Our next demonstration (October 1998) will use the vision module (Section 3.1) to accomplish the same end.

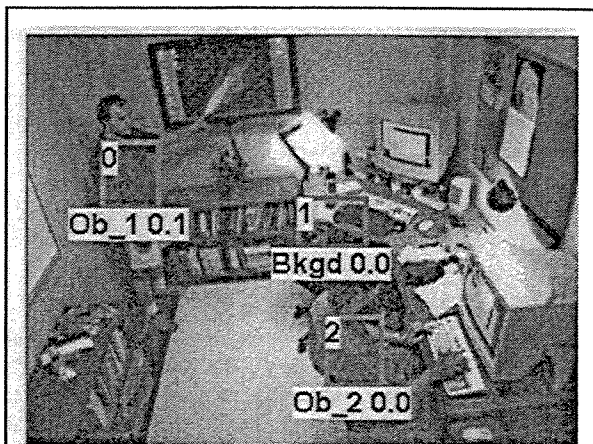


Figure 2: A view of the office from one camera, with three "hotspots". The user interface migrates between the three screens (including the large one on the back wall) as the user moves from hotspot to hotspot. Based on color histograms, the program has identified the contents of the hotspots as object 1, background, and object 2.

Using stereo and color together, we will track users instead of relying on hotspots.

4.2 Contact Anyone Anywhere

We will demonstrate several EasyLiving capabilities with our "Contact Anyone Anywhere" demonstration in April 2000. In this scenario, a user in an EasyLiving space will signal that she wants to place a call to someone else in an EasyLiving space. This signal may be given using a keyboard, mouse, voice, or gesture command. Since EasyLiving will know the user's identity, it will present her personal phone list, and the user will indicate which person she wants to call, using one of the same set of command modalities. EasyLiving will place the call, signaling the callee in the appropriate place and in an appropriate way. During the call, the user will switch to another UI device, move to another room, and finally pick up the call on a mobile device, all while maintaining the conversation.

4.3 Child Care Assistant

One attractive application of EasyLiving is to aid in caring for a child or a pet. EasyLiving could act as an enhanced child monitor, checking for dangerous conditions (e.g. near top of stairs), monitoring protected spaces, information, and vital signs. The system could also monitor a babysitter's time with children. If any condition required attention from a parent, the system could notify them in an appropriate way, including making a cellular telephone call. We plan a demonstration of these ideas in April 2000.

4.4 Vision-Based Home Automation

Having cameras in the room invites many interesting applications. For instance, cameras could make a video history of the space, recording during those times when motion occurs. The video history could be used to answer questions of the type, "What happened?" For instance, "Where did I leave my keys?" "What did the burglars look like and what did they take?" "How long has that vase been missing?"

Cameras could also be used to adjust light levels appropriately. If a person starts reading, the cameras can measure the ambient light and adjust lamps until the light is bright enough. Used as motion detectors, the cameras could cause the system to turn off lights in unoccupied rooms.

5. Summary

EasyLiving is looking beyond the desktop as the next step in computing for the everyday user. By connecting the home and work environment to the PC, we can provide casual access to computing. We are building an architecture and technologies to investigate and demonstrate this concept.

Acknowledgements

Thank you Pierre De Vries, Director of the Advanced Products Group at Microsoft for helping to get EasyLiving started from the beginning.

Thank you also to Charles P. Thacker, Director of Advanced Systems at Microsoft Research in Cambridge, England for contributing concepts to EasyLiving, in particular the ideas of self-aware spaces and casual access to computing.

References

- [1] M. H. Coen, "Design Principals for Intelligent Environments," presented at AAAI Spring Symposium on Intelligent Environments, Stanford, CA, 1998.
- [2] M. Lucente, G.-J. Zwart, and A. George, "Visualization Space: A Testbed for Deviceless Multimodal User Interface," presented at AAAI Spring Symposium on Intelligent Environments, Stanford, CA, 1998.
- [3] M. J. Swain and D. H. Ballard, "Color Indexing," *International Journal of Computer Vision*, vol. 7, pp. 11-32, 1991.
- [4] K. A. Bharat and L. Cardelli, "Migratory Applications," presented at UIST '95, Pittsburgh, PA, 1995.
- [5] S. Oviatt, "Multimodal Interactive Maps: Designing for Human Performance," *Human-Computer Interaction*, vol. 12, pp. 93-129, 1997.

Synthesized Multimodal Information Spaces With Content-Based Navigation

Joseph Sirosh and Marc Ilgen

HNC Software Inc.
5930 Cornerstone Court West
San Diego, CA 92121
{js,mri}@hnc.com

ABSTRACT

In this paper, we develop a vision for a smart-space based information management system, and discuss the scientific and technical challenges that need to be overcome to make it a reality. Perhaps the most challenging technical problem is efficient handling of unstructured multimedia data. How might one index and retrieval multimedia data? How could one extract high-level information from complex sensory inputs? How might one integrate information across different media and construct synthesized percepts? How might one encode cross-media associations and content? This paper will outline some of these challenges, and suggest approaches that hold promise for tackling these problems.

1. INNOVATIVE CAPABILITY ENVISIONED

Imagine a personal computer with a video camera, a microphone and assorted sensors such as touch-pads or virtual reality gloves. Imagine being able to show a picture to the camera and have it retrieve, from a huge multimedia database, pictures that look like it, text documents that describe it, web pages with similar content, and even related audio clips. Imagine being able to play back a piece of audio, e.g. a conversation in an unknown language, or a piece of music, and being able to retrieve audio clips that sound similar, and even multimedia web pages and descriptions. Now imagine that this approach is generalized to any form of information, e.g. those coming from battlefield sensors. This is the fundamental capability we envision.

In more formal terms, we envision a truly heterogeneous information space which holds knowledge in multiple modalities: visual, auditory, other sensory, and textual; and one in which the information is associated and linked across the multiple modalities based on content. Two properties of the system are important: the fact that it performs cross-modal synthesis of information, and that it learns actively instead of being a passive, fixed system operating solely on pre-recorded knowledge. Without cross-modal synthesis of information, a smart space might be just a fancy database: but with it, the system will be imbued with rich, content-based associative links that can be explored by humans and automatic systems. Objects in this integrated information space will be implicitly

or explicitly associated with attributes in a variety of media: for example, an aircraft might be associated with pictures of aircrafts in decreasing order of resemblance, textual descriptions, sound and video clips in decreasing order of relevance, and infrared or radar signatures also ordered by resemblance. Many of these associations will be implicit in the representation of the object itself, but others will be explicitly represented and coded. Most of the associations will have been automatically extracted in learning processes by frequently seeing co-occurring events in the information space, much as humans associate events. Others would have been produced by active exploration of networked databases and other resources such as the Internet, perhaps using intelligent agents spawned off by the system in response to user queries or other triggering events. The statistical associations produced and stored as a byproduct of these learning processes would gradually accumulate to form useful knowledge that can be called upon by humans and expert systems to synthesize new, heretofore unseen scenarios and plausible events. Thus, the smart space will also be an active, learning system that not only collects and organizes data, but also records and exploits the associations between data attributes and between different multimodal objects in the database. Through constant use, the system will become further enriched as it discovers new statistical associations and relationships among the data through its ongoing learning processes.

2. MAJOR TECHNICAL CHALLENGES

To realize smart spaces that are practically useful to defense and intelligence analysts, one must combine two types of technologies. The first consists of infrastructural technologies, such as special sensors and display devices, novel user interfaces, mobile networking methods, visual, auditory and gesture recognition technology and intelligent animation technologies. These form the "hardware" from which the smart space is built. The second set of technologies that are at least equally, if not more, important to the analyst are the multi-modal, intelligent information management algorithms that work seamlessly with the smart space infrastructure to allow analysts to transform data into useful knowledge, to collaborate and to visualize information, to intelligently retrieve data and perform value based filtering, to fetch, create and express scenarios, and to develop a high-level intelligent

awareness of dynamic situations. These form the core “intelligence” of the smart space. The rest of our paper will be focused on this issue.

Information management for smart spaces is highly challenging, scientifically and technically. Yet, clearly, here also lies the potential for greatest reward. An integrated smart space information management system would present a revolutionary advance in our ability to rapidly access, analyze and interface with vast quantities of unstructured information, especially in time-critical situations. The technology could be an integral part of pursuits as diverse as battle management, intelligence analysis and electronic libraries scattered across the Internet and intranets. Some of the key technical challenges to be overcome first are:

1. Effective algorithms for indexing and retrieval of multimedia information.
2. General algorithms to extract high-level information from arbitrary sensory modalities.
3. Algorithms to synthesize information across modalities and create multi-modal associations.
4. Algorithms to compactly encode and store information across modalities.
5. Efficient, scalable implementations for terabyte.

Each of these is described below.

2.1. Multimedia Indexing and Retrieval

A key challenge for smart spaces is to be able to effectively index and retrieve multimedia resources such as images, video and audio, and associated text. Smart spaces may also require one to index and retrieve human gestures and related sensor data, facial expressions, verbal commands and animation sequences. How could one handle a multimedia document containing such complex data as database objects? What form should the database take? How could one index it by content in a fashion that permits retrieval by queries in a variety of media, such as other novel images or video clips or sounds, gestures or facial expressions? Can one develop media independent techniques for indexing such data?

2.2. Extracting High-Level Information

Another important challenge for a smart space information management system is the development of algorithms to extract high-level information from multimedia data without human intervention. Such a capability would permit the smart space to automatically explore new information sources, collect relevant information, and perform value-based filtering. Extracting high-level information has also been the goal of

research in image understanding and computer vision for several decades, but traditional image understanding algorithms have proven to be too brittle and/or too limited in domain to be of practical utility in many problems of practical interest. However, recent fundamental advances in neuroscience and our understanding of the human brain at a computational level have given us sufficient insight to design approximate algorithms that work over a very broad domain, at the expense of some accuracy. Novel discoveries from neurobiology are leading to the synthesis of algorithms modeled on the dynamics of neural activity that are significantly different from traditional image processing and statistical algorithms. There is strong reason to believe that an interdisciplinary research effort, synthesizing the emerging understanding of the human brain from neuroscience with computer algorithms and human-centric interfaces can lead to important breakthroughs in tackling this problem.

2.3. Cross-Media Content Synthesis

While content extraction algorithms developed for specific modalities would be useful in and of themselves, additional algorithmic advances are required to enable cross-media content characterization in a unified manner. Knowledge in the natural world is an integrated sum of data in visual, auditory, textual and other sensory forms. Human perception of events and objects are not merely a collection of images or sounds, but an integrated representation at a higher, synthesized level. Is it possible to achieve such a cross-media, synthesized representation of information? How could we establish associations between data in multiple media so as to represent events and scenes? How can we index such information? Can we exploit cross-media associations to identify and analyze situations and their unique or common characteristics, and to classify them? Can such learned associations be used in creating scenarios for smart interactions among analysts?

2.4. Compact Cross-Media Encoding

To effectively manage the vast quantities of digital information that will have to be managed in Smart Spaces, it will be necessary to encode information content in very compact form. In effect, the key semantic features of each document, image, video, or audio clip must be stored in a manner that minimizes the number of bits used without reducing the value of the information. This compact form must be applicable to information across modalities. Current methods for information storage are intimately linked with the underlying modality: words and phrases for documents, image features for images, etc. Algorithms that transcend these media boundaries must be developed. Compact encodings are also crucial in smart spaces if any form of distributed processing or collaboration are involved, since limited communication bandwidths are likely to be a very serious factor affecting performance.

2.5. Efficient, Scalable Implementations

Much of the past work in handling multimedia information have resulted in "toy" systems that work for specific domains and limited data, but are not scalable to larger problems. For example, even the best systems for Automated Target recognition (ATR) are only capable of discriminating between a small number of possible target types. In the constantly evolving world of digital information, such domain-specific methods are of very limited utility. Algorithms must be developed for identifying and encoding multimedia content in such a way that the system can scale up over time to store, represent and encode huge amounts of data, from a wide variety of domains. These algorithms should be based on continuous, on-line learning techniques, since simple *a priori* specification of classification and organizing methods for unseen data is likely to be ineffective. Can we create general multimedia database systems with on-line learning and indexing algorithms that are scalable and can store vast amounts of information?

3. PLAUSIBLE APPROACHES

Over the past few years, researchers at HNC Software Inc. have been developing several technologies that provide partial solutions to some of the problems listed above. Many of the technologies build upon the core competencies of the company in computational intelligence, statistical methods for information representation and retrieval, learning algorithms, image processing systems for automatic target recognition and intelligent agents. Also, several novel tools for multimedia processing, based on algorithmic insights gleaned from recent computational neuroscience research are being developed. Some of these technologies will be discussed in brief below.

Two technologies form the cornerstone of our approach to media-independent, content-based representation and processing. The first is a novel unsupervised learning principle called LISSOM, derived from interdisciplinary research on brain function. This principle is key to extracting "content" as represented by higher-order statistical structures in the input. The second is the Context Vector technique for compact vector-space representations of co-occurrence information. This technique is key to forming compact, efficient and scalable representations of information.

The LISSOM model was originally developed by Sirosh (1995, 1998) as a computational model of the primary visual cortex. LISSOM tackles what has been for a long time a "holy grail" of unsupervised learning, namely the discovery of a general self-organizing principle which could be applied in multiple stages to discover increasingly complex structures in arbitrary input data sets. It turns out that this principle is also key to how the primary visual cortex of the brain develops after birth by seeing visual input, and organizes the

structures necessary for vision. The LISSOM algorithm has been successful in modeling this development, and in addition could computationally predict structures in the brain that neuroscientists had not observed previously, but were subsequently verified. Essentially the algorithm reduces visual or other inputs to high-level statistical structures, thereby forming a set of complex feature detectors for the underlying information. These feature detectors form a highly information-rich, transformation invariant, statistical vocabulary for representing multimedia data. Currently, this technology is an important part of a DARPA-sponsored project on Collaboration, Visualization, and Information Management, currently underway at HNC.

The Context Vector technology (Caid and Carleton 1994) was originally developed as a compact vector-space representation for text documents. By encoding statistical co-occurrences of words in text, the technique provided a means to perform intelligent text retrieval and routing based on content. It was soon discovered that the same technology provided a powerful, domain-independent means to represent content in non-textual media as well, provided a vocabulary of highly informative features could be extracted in a preprocessing step. The context vectors would then encode the statistical co-occurrence information between elements of this feature vocabulary and enable compact indexing of content. Based on this idea, HNC developed the ICARS image retrieval system (Pu and Ren 1995) using wavelet feature detectors and context-vectors. Although the system could not exceed the performance of more specialized and highly tuned image retrieval systems, it gave the capability to automatically index a wide variety of image data and perform approximate retrieval with very low levels of effort. The key limiting factor in the system was the problem of extracting and representing complex feature detectors for images. With the development of the LISSOM algorithm, this limitation is currently being transcended. Together, the LISSOM and Context-Vector techniques being developed now will provide the capability to extract high-level information from multimedia data, create compact content-based indexes and perform intelligent multimedia retrieval.

Researchers at HNC are also beginning to explore new techniques for cross-modal synthesis based upon LISSOM and context-vector techniques. The basic idea is that cross-modal associations can be learned from the co-occurrence of high-level features extracted by LISSOM from each medium. This co-occurrence information can be encoded in context vectors as well, and later used to create compact, higher-level indexing systems that can represent not only the content of a single image or piece of text, but a whole multimedia document. The cross-modal associations also enable cross-media retrieval, making it possible, e.g. to retrieve images using text queries or audio clips using image queries.

Another research effort currently underway at HNC, funded by the DARPA Hybrid Information Appliances program, is exploring sparsely coded models of cortical information processing to develop efficient methods for representing and processing sensory information and interfacing with external systems. Yet another research effort is addressing distributed information retrieval and collection using autonomous neural agents. This agent-based approach could in principle provide a scalable, fast system for information gathering, collection and organization in a distributed network of computers that are linked to a smart space.

In our future research, we hope to address several new issues that are directly relevant to smart space information management systems. We hope to extend our multimedia information management systems to arbitrary sensor data. We also plan to develop systems to (1) facilitate active exploration of distributed information sources, (2) perform intelligent monitoring of key data sources, (3) do automatic value-based filtering and relevance determination, (4) perform event-detection and raise alerts, and (5) interface with more sophisticated automatic reasoning systems in order to automate or facilitate the process of complex decision making in a smart space.

4. POTENTIAL COMMERCIAL APPLICATIONS

The commercial applications of this technology are numerous. One important example is a worldwide, active digital library that transforms the world-wide web into a true multimedia information space that can be navigated with virtual reality tools. More mundane applications include multimedia databases, automatic routing of multimedia documents based on content over electronic networks, multimedia chat rooms and collaborative spaces. The technologies developed for such a system would also permit a variety of ways of visualizing information, and comprehending raw unstructured data using the associative links that get established. For example, it might become possible to describe a new image in text and audio by finding what other images, audio clips and textual documents it might be associated with. We also believe that these technologies would lead to intelligent human-centric interfaces to computer systems that make it possible to establish far more efficient ways of retrieving information and synthesizing scenarios and customized sub-domains of information spaces.

5. CONCLUSIONS

Integrated smart space information management systems makes it possible for humans to interact with vast quantities of information using our highly evolved, massively parallel sensory systems. It permits a new level of situation awareness and decision making capabilities by giving the user the ability to rapidly access, understand, summarize and convey

information in multiple forms and from a variety of sources. To construct such a system, one must tackle challenges such as scalable multimedia databases, content-based indexing and retrieval of multimedia, effective techniques for extracting high-level information from raw sensory data, cross-modal synthesis of information, and compact encodings of content. We believe that some of the key challenges in handling multimedia and raw sensory data can now be overcome using algorithmic insights derived from recent research in neural information processing. Using these insights, it should be possible to construct smart space information management systems that continuously learn and actively explore the electronic world, and perhaps pave the way for a new paradigm in information processing.

References

Caid, W., and Carleton, J. (1994). Context vector based text retrieval. In *Proceedings of the IEEE Dual-Use Conference*.

Ilgel, M. and Rushall, D. (1996). Recent advances in HNC's Context Vector information retrieval technology. In *Tipster Phase II 24-Month Meeting*, Tysons Corner, VA, 1996.

Maybury, M.T. (1997). *Intelligent Multimedia Information Retrieval*. AAAI Press/MIT Press.

Pu, K., and Ren, C. (1995). Image/text automatic indexing and retrieval system using the context vector approach. In *SPIE*, vol. 2606.

Sirosh, J. (1995). *A Self-Organizing Neural Network Model of the Primary Visual Cortex*. PhD thesis, Department of Computer Sciences, University of Texas, Austin, TX. Technical Report AI95-237.

Sirosh, J. (1998). A hierarchical algorithm for unsupervised identification of nonlinear manifolds. In *Proceedings of the 5th Joint Symposium on Neural Computation*. La Jolla, CA: Institute for Neural Computation, University of California, San Diego.

The Dynamic Human Form: Wearability Issues Revealed!

John Stivoric, Chris Kasabach, Francine Gemperle, Malcolm Bauer, Richard Martin

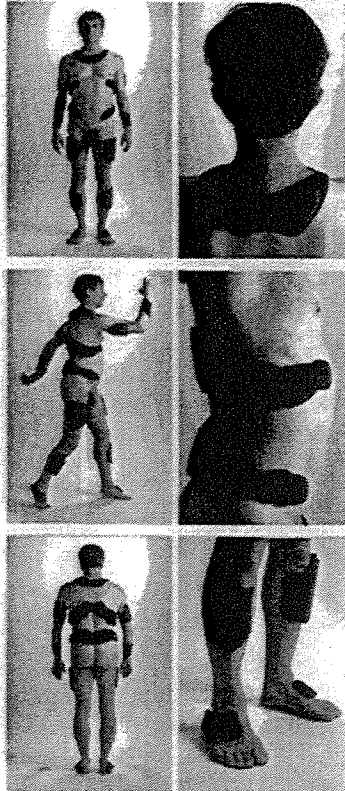
EDRC* at Carnegie Mellon University

Pittsburgh, PA 15213 USA

+1 412 268 7890

{kasabach, stivoric}@cmu.edu

<http://www.edrc.cmu.edu/design>



ABSTRACT

This paper explores the concept of dynamic wearability. Our research has been to locate, understand, and define the spaces on the human body where solid and flexible forms can rest without interfering with the various movements of the body – including skeletal and muscular movements and the fluctuating sizes of human fat deposits. The result is a set of design constraints embodied as wearable forms that define the available spaces on the body for the design of comfortable, manageable and unobtrusive wearable computer systems and other body worn devices.

Keywords

dynamic wearability, design spaces, design research, wearable computers, user-centered

INTRODUCTION

Computing is no longer limited to the office desktop and its system components – monitor, keyboard, mouse, and windows interface. Pagers, cellular phones, wearable computers and other small, mobile computing devices are becoming more prevalent.

The design constraints for mobile and wearable computers are both numerous and varied. One level of constraints stem from user needs, human perception, the physics of the human body, and the task being performed. On top of these issues are technological constraints including hardware, software, thermal, and packaging limitations.

In this paper we address the first level of constraints. We present an initial set of wearable forms that attempt to map the most unobtrusive design spaces on the human body in motion. These forms and locations take into account a wide range of tasks, motions and body types, as well as proportion differences between adult men and women.

This exploration grows out of understanding gained from over 7 years of hands-on, in-the-field experience working with people in developing mobile and wearable products for a variety of industrial and commercial applications [21].

PROBLEM

As designers, we have not found an extensive resource that adequately maps 3D spaces on the body. While there are some excellent human factors reference tools available to designers such as *The Measure of Man and Woman* [2], and some more complex references available through the United States military forces, these generally define the static and linear dimensions of the human body. These references provide a starting point for the issues of wearability but do little to aid us in understanding moving organic surfaces of the body in motion – how the surface changes and folds as we twist, bend, crawl, climb and reach.

METHOD

The method we used to discover locations and forms for the human body was an iterative one that employed both two dimensional drawings and three dimensional foam models. We began by locating the areas on the body that are relatively large, immobile while in motion, and similar in size across adults. We then sketched over various drawings and photographs of the human body to understand how these areas were further carved by muscles, fat, and skeletal movement. From our drawings we made three dimensional foam models and put them on a wide range of human volunteers. Then we annotated and carved the foam models based on user feedback. From there we put the forms into CAD software and designed the next iteration of foam models to test.

RESULTS

The results are a series of wearable ‘pods’ and flexible forms encased in minimal, stretchable fabric structures. The pods are smooth and largely organic with scallops and arcs to allow the major motions of the limbs and torso as well as muscle flexion and fat deposits. In some cases the pods are large and rest by themselves. Where more motion or fluctuation between body sizes was noticed, the pods tend to become smaller. Linking many small pods together with flexible fabric gaps in between provided for the greatest amount of versatility and comfort.

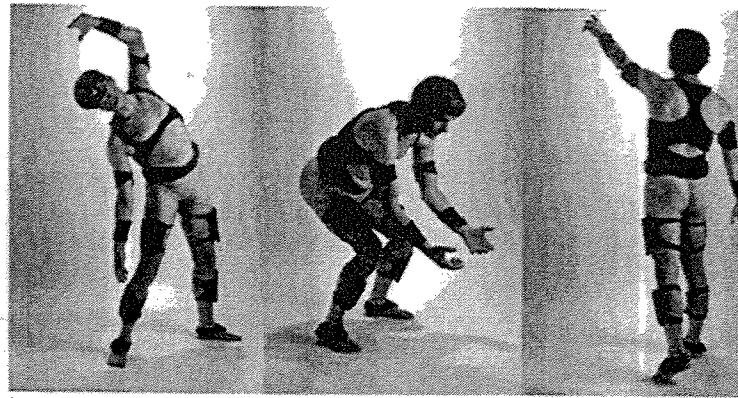


Figure 2: Forms in stretchable fabric structures

CONCLUSION

Our results are an initial effort to define dynamic wearability. We intend to conduct more extensive user testing to verify these results through formal evaluation techniques. The final result will be a series of clear documents that present both a broad set of suggested guidelines and a dynamic human factors chart for designing comfortable, manageable and unobtrusive wearable products.

ACKNOWLEDGMENTS

We would like to thank Dan Siewiorek, EDRC at Carnegie Mellon*; Dick Urban, DARPA for his continued support, Dave Alberti, and Lorna Ross-Brook of LRB Design Research.

*The EDRC resides within the Institute for Complex Engineered Systems (ICES).

REFERENCES

1. Bass, L. et al. *The Design of a Wearable Computer*. CHI97
2. Tilley, Alvin R. *The Measure of Man and Woman*. Henry Dreyfuss and Associates, NY.

©1998 copyright on this material is held by the authors

Adaptable Protocols for Smart Spaces

Mitchell Tsai, Mark Yarvis, Peter Reiher, Gerald Popek

Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095
tsai@ucla.edu
<http://fmg-www.cs.ucla.edu/>

ABSTRACT

Human languages have an adaptability and resilience far surpassing current electronic protocols. Techniques based on human dialogue can assist in solving protocol creation, translation, and update problems for long-lived smart space devices. Devices will probably exist within environments which are similar to human communities, where communications are often facilitated by third parties, regulatory committees, and information repositories, but where external mediation is sometimes unavailable. Rather than relying solely on pre-tested protocols and secure transmission, smart devices can internally test protocols for faults, borrowing techniques from existing hardware and software simulation languages and from Artificial Life simulation environments.

1. ADAPTIVE COMMUNICATION

Adaptive communication between electronic devices must recognize and respond to dynamically changing relationships among a wide variety of devices. For systems which can exist and function over decades, communication protocols should be able to automatically upgrade, even handling devices which do not yet exist. Also, we would like to minimize the massive maintenance and upgrade costs of current-day computer software.

One innovative approach is creating adaptable protocols which automatically discover and learn by trading agents which operate in internally simulated environments within each device. When no common protocols exist which are powerful enough for the desired communications, these agents try to establish communication within simulated environments, trading meta-information directly when possible, and communicating final results back to the original device.

Current communication protocols suffer from many limitations which do not exist in human languages. Consider modem communication. There are some limited negotiations when two modems communicate to adjust bandwidth and compression, but the protocols are preloaded onto modems and suffer the following problems:

- Incompatibility - When two different protocols are used, the result is usually complete miscommunication. Switching protocols may be possible, but only if they are

preloaded. Even with the same protocol, many parameter settings must be correctly aligned beforehand, or communication will fail.

- Upgrade problem - Devices can not discover incremental changes in protocols. Instead, protocols must be completely replaced through software or hardware upgrades.
- Centralized protocol creation - Protocols are created off-line by humans. There are no means for individual devices to negotiate and create new protocols when current ones fail. When standards for new communication speeds are still being negotiated, manufacturers produce modems which must be upgraded when final standards are set. If two devices meet and one requires IR communication but the other requires wireless radio, successful communication may require third-party negotiation and the creation of hybrid protocols.
- Internal faults - There is no means for correcting a protocol which is inherently faulty or lacks the necessary capability to communicate with a new device which can not be described by the current device interface standard.

Human language systems have strong learning capabilities. New words are discovered and learned. New meanings can be learned for old words. When two people who speak different languages encounter each other, they can shift to a common language. If no common language exists, they create a *pidgin* language for temporary communication. If people continue using the *pidgin* language, it begins to take on the full power of a permanent language, developing into a *creole*.

In contrast, device languages and computer standards tend to be pre-designed, fairly limited, and overly long-lived (e.g. IPv4 to IPv6).

A typical approach is for a "far-seeing" company or committee to design a communication standard, with some means for exporting device interfaces. An attempt is made to anticipate all possibilities, incorporating the reasonable ones into the standard. The communication protocol is completely analyzed beforehand, so that it satisfies efficiency, correctness, and fairness requirements for the target environment. The decided

standard is often used longer than desired, since the cost of switching to new standards is often very high.

Object-oriented languages and component-based communication systems provide some adaptability [1, 2], but "The current general way to describe interoperability information is to provide the component's interface definition and some additional informal documentation. This level of information is obviously not detailed enough to manage interoperability issues on a reasonable level... At the moment, a general established technique for the complete description of semantic interoperability of components does not exist." [3]

"Systems such as CORBA, OLE, and COM define the interfaces they provide to other components, but not the interfaces they require from other components." [4] They also don't provide a mechanism for creating new interfaces and updating old interfaces. Interfaces may need to be established in a two-way process. Dynamic linking is just a start towards fully dynamic communication [5, 6].

2. POSSIBLE APPLICATIONS AND BENEFITS

Communication in simulated environments can occur much faster than real world communication, so protocol discovery can occur without tying up real-world bandwidth.

With adaptable protocols, groups can independently develop mobile computers and intelligent objects before worldwide standards are established. Two-way definition of interface languages provide extended flexibility for future environments.

3. MAJOR TECHNICAL OBSTACLES

We want introspection, inspection, and communication methods which can establish initial communication with foreign devices, as well as trade the meta-information necessary for full communication. Our adaptable protocol system should be robust enough to redesign itself in all major areas, while maintaining self-check mechanisms and safety fallbacks to survive crashes and problems.

4. PLAUSIBLE APPROACHES

4.1. Initial Communication

Some basic signals should be semi-reserved for "SOS" and "Query" commands, but these can also be flexible. For example, human languages don't even have the same words for "help", "yes", and "no." There is no *a priori* need for computer languages to have identical ones either. Devices can help each other learn the basic communication frequencies, encodings, signals, and words through Artificial Life techniques [7] and pattern detection algorithms.

Basic communication can be established using a base-level fallback protocol which can establish communication at the signal processing level. As long as electronic signals can be sent, analog or digital, a fallback protocol can allow

communication from the level of raw signals up to protocol naming and identification; i.e. "Can you speak protocol XYZ?" The fallback protocol could investigate line quality and available communication speeds, just as modem protocols do today.

Existing protocols and/or a protocol manager must distinguish noise and errors from attempts to communicate using a fallback protocol.

Once bare minimum communication is established, lists of protocols or interfaces can be exchanged. However, devices might have no mutual protocols powerful enough for the communication they need. A knowledge exploration system (using decision trees, Bayesian networks, expert systems, parallel simulated search etc...) can guide devices through the proper questions to ask when learning the details of a new protocol, or when learning modifications to an existing protocol.

Also, an intermediate device (or devices) may be needed to establish communication between two devices which don't have a mutual communication channel (i.e. one speaks IR, but the other uses RF) with suitable bandwidth (i.e. both have IR, but they have mutually incompatible RF which can be handled by an intermediary).

4.2. Protocol Learning in Simulated Environments

Protocol learning and adaptation can be done through the trading of agents, rather than communicating over electrical or electromagnetic space, especially in insecure or noisy environments. The communication in simulated environments can take place much faster than real-world communication.

After initial communications are established, agents can be traded between devices. The agents can behave like the originating devices, learning and adapting protocols in simulated environments (which each device maintains), communicating the final results afterwards.

Some simulated environments can be very basic, others very complex. The protocol learning may occur in stages for very complicated languages, as communication may not just be "turn on the light", but may involve migrating InterJava 9.4 code. Initially a simple agent maybe sent for simulation environment A-1, but after one pass of negotiations, a more complex agent may be sent for simulation environment B-5.

4.3. Redesigning Communication Languages

Computer communication languages must develop the open-ended queries which are possible in human languages (and allow for their great adaptability). We can model the creation of new and hybrid protocols on human language discovery, rather than use fill-in-the-blank models of object properties and methods. Developing fill-in-the-blank templates are great, but we need methods to overcome their limitations, as well as methods to create templates when they don't already exist.

Open-ended and specific queries should be an integral part of the communication languages. "What does X mean?" "I am

speaking a local version a7 of XYZ 1.32. Which version of XYZ 1.32 are you using?" "What are the differences between our versions of XYZ 1.32?" "Help!" "What are you doing?"

Languages should have response systems that detect attempts to communicate, distinguishing them from actual communication using known protocols. Basically, meta-questions and meta-answers.

4.4. Third-party (Network) Assisted Communication

Version control and replication systems [8, 9, 10] are needed to maintain and spread knowledge of multiple communication protocols. Individual devices do not need to know all protocols, but they should keep knowledge of the ones they are most likely to encounter.

Also, they may need to contact central repositories or other devices to learn (or relearn) protocols [11]. "I have contacted some device speaking a WX-type language, but have deleted my WX languages and can not figure out how to communicate with the WX device. Can you send me the latest WX negotiation protocol? I am having trouble with the encoding scheme."

"Device X has a 'verify' operation which I do not recognize. What is it? Can I reproduce it using my 'read' and 'write' operations?"

4.5. Internal Protocol Testing

Individual smart devices can test protocols for faults in internal simulations, borrowing techniques from hardware and software simulation languages (i.e. Maisie) [12] and Artificial Life [13] simulations. Rather than relying solely on pre-tested protocols and secure transmission, devices gain the flexibility to do partial or total internal protocol design with some protection, when communication might be impossible with existing well-tested protocols.

Secure codes will still be useful for transmitting worldwide or large-area updates to major communication protocols, and trusted repositories and other devices will use various levels of security methods in ordinary communication.

References

1. Bosch J., "Language Support for Component Communication in LayOM", *10th Annual European Conference on Object-Oriented Programming (ECOOP '96)*, July 1996.
2. Weck W., "Independently Extensible Component Frameworks", *10th Annual European Conference on Object-Oriented Programming (ECOOP '96)*, July 1996.
3. Murer T., Scherer D., Wurtz A., "Improving Component Interoperability", *10th Annual European Conference on Object-Oriented Programming (ECOOP '96)*, July 1996.
4. Olafsson A., and Bryan D., "On the Need for 'Required Interfaces' of Components", *10th Annual European Conference on Object-Oriented Programming (ECOOP '96)*, July 1996.
5. Diniz P. and Rinard M., "Dynamic Feedback: An Effective Technique for Adaptive Computing," *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI '97)*, Las Vegas, NV, June 1997.
6. Fisher J., "Walk-Time Techniques: Catalyst for Architectural Change," *Computer*, September 1997, pp. 40-42.
7. Dyer M., "Grounding Language in Perception," *Artificial Intelligence and Neural Networks: Steps toward Principled Integration*, eds. V. Honavar and L. Uhr, Academic Press, NY, 1994, pp. 455-482.
8. Ratner D., "Roam: A Scalable Replication System for Mobile and Distributed Computing," UCLA Computer Science Department Ph.D. Dissertation, UCLA Technical Report CSD-970044, 1998.
9. Reiher P., Page T., Popek G., Cook J., and Crocker S., "Truffles - A Secure Service for Widespread File Sharing," *Proceedings of the Privacy and Security Research Group Workshop on Network and Distributed System Security*, February 1993.
10. Reiher P., Popek G., Gunter M., Salomone J., and Ratner D., "Peer-to-Peer Reconciliation-Based Replication for Mobile Computers," *ECOOP '96 Second Workshop on Mobility and Replication*, July 1996.
11. Alwan A., Bagrodia R., Bambos N., Gerla M., Kleinrock L., Short J., and Villasenor J., "Adaptive Mobile Multimedia Networks," *IEEE Personal Communications*, April 1996, pp. 7-22.
12. Liu W., Chiang C., Wu H., Gerla M., Jha V., and Bagrodia R., "Parallel Simulation Environment for Mobile Wireless Networks," *Proceedings of the 1996 Winter Simulation Conference WSC '96*, eds. J. Charnes and D. J. Morrice, Coronado, CA, 1996.
13. Werner G. M. and Dyer M. G., "Bioland: A Massively Parallel Simulation Environment for Evolving Distributed Forms of Intelligent Behavior," in *Massively Parallel Artificial Intelligence*, eds. H. Kitano & J. Hendler, AAAI Press / MIT Press, Menlo Park, CA, Chapter 10, 1994, pp. 317-349.

ACTIVATING EVERYDAY OBJECTS

Roy Want, Mark Weiser
Xerox PARC
3333 Coyote Hill
Palo Alto, CA 94304
[want, weiser]@xerox.com

Elizabeth Mynatt (affiliation as of 9/15/98)
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280
mynatt@cc.gatech.edu

1 INTRODUCTION

The confluence of technological advances in wireless technology, mobile computing, novel displays, and sensors in addition to the decreasing cost of computing power provides the opportunity for utilizing computational capabilities in an increasing number of specialized, yet networked devices. By integrating and embedding these devices into “smart spaces,” we have the opportunity to move computing power from its specialized location as a desktop PC to distributing computational capabilities into the existing infrastructure and tools of everyday offices. In this paper, we discuss the inherent advantages of enhancing everyday objects with computation, describe three key research problems that must be addressed to achieve these advantages and outline our approach in these areas.

People work in the everyday world. We use, when we are at our most effective, those tools and methods with which we have long experience and familiar skill. These include, for many people: pencils, paper, whiteboards, desktops, paperclips, notepads, and the many other objects of everyday office life. Although the WIMP metaphor tries to replicate these items in a virtual space, it fails to capture their ease of use, their flexibility, and their serendipity.

Instead of using a metaphor, can we activate the everyday objects in the world? Yes. Work in Ubiquitous Computing [5][6][7][8], Augmented Reality[1][3], and Tangible Bits [2] has shown the way. Yet work in all three areas has fallen short of what they hoped to achieve. In all of these areas three key research problems show up again and again, and without much progress. Until these problems are

solved, real applications will be difficult to achieve. The three key problems:

- Short-range, in-building location. For example, the most widely used research in-building location system (developed by one of us) is the Olivetti active badge system [4], used at Xerox PARC and elsewhere. But its spatial granularity is tens of feet, and its temporal granularity is tens of seconds. These are values ill-coordinated to human scales of fractions of inches and fractions of seconds which is important for many applications the badge system was not designed to support.
- Coordination of active objects into a single UI. For example, a room may contain a digital camera, a PC, a video projector, an active pen-input screen, a microphone, a printer, a scanner, and several palm pilots. To the human in the room, who can see them all, it is obvious that they form a set of resources that can be coordinated to get work done. But the view from inside the microprocessor in each of these devices is of isolated worlds-in-themselves. How can we bring the machine viewpoint in line with the human?
- Coordination of real and virtual objects. For example, a document that is scanned, printed, emailed, stored in different real and virtual files, pasted on a wall, etc. etc. is still the same document from a human point of view. We think of it as “the same”. But to our machines most of these versions don’t even exist (e.g. the one pasted on the wall), and the rest are not seen as

even related. Can we bring the machine's understanding of relationships, even such a fundamental relationship as equality, in line with the human?

The advance we envision is bring the technical elements of everyday technology in line with the human uses of that technology. The result will be a significant increase in the impedance match between machine and human, leading to much more effective work.

2 OPPORTUNITIES

We envision applications within the everyday paper-based work of government and military offices. The individuals in these offices use many sorts of everyday objects in special and specific ways. The placement of items, as well as the content of items, in these offices is crucial for getting the work done.

The measurable challenges are:

- To increase the flow of decisions and accurate information in the office.
- To decrease the number of mislaid decisions, information, or workflow.
- To fit into the everyday office practice, without requiring significant retraining.

To meet these challenges, we intend to enhance and connect interaction with various individual objects in the office-place. By augmenting common objects, such as a whiteboard or notepad, we leverage everyday office practices. By connecting these devices, we create a networked web of information that can be accessed throughout the office in contrast to requiring access at the original input source.

Once this web of information and devices is in place, the user's model of interaction changes from an application-task based model to an information-flow based model. Since many office activities are centered around information management, our new UI paradigms must adopt an information-centric point of view. For example, instead of producing a design specification document for a piece of software, the activity to be supported is managing the design process and the information associated with that process. That information will appear in many places and formats, such as the project plan, within code itself as well as current design specs.

3 TECHNICAL CHALLENGES

In the area of location, there are many technical methods of doing the job. They are all too expensive, or too prone to interference in an office environment. GPS does not have the resolution and is not easily accessible from inside a building. Luckily, there are many possible approaches to resolving the problem. Multiple-camera data fusion, sonic locators, and passive antenna-detuned triangulation are a few of these. The key challenges are low cost, and effective reliability. The standard to beat is the eyes and hands of the person in the office, who rarely fails to reach out and grasp the document once it is in front of him. Our location technology must do as well.

In the area of UI coordination, there are two challenges. First, it is to understand the user's expectations of the linkage of the office objects. That is to say, what is the semantics that the user imputes to the objects. Secondly, it is to deliver on that imputed semantics. This is a challenge that admits many kinds of solutions, with deep challenges in system integration. For instance, when devices with displays are close enough together having them share parts of the same image. In the end, the challenge is one of design: will it all hold together for the user.

In the area of coordinating real and virtual, the technical challenge is labeling. How can the many versions of the "same" thing be labeled as the same, especially if some of them only have physical manifestations? Supposed they all had barcodes? How would the barcodes be applied, when, how would they be read, how would they be looked up, in what database, controlled by who, accessible where and when? Are barcodes the right thing? Should we consider RFID-tags or other electronic tagging technology?

4 TECHNICAL APPROACHES

4.1 Location on a Planar Surface

In an office environment many aspects of our work are focused on the desktop or the whiteboard. If objects in these areas can be identified along with their location, it is possible to enhance work practice by further augmenting the space. We have considered a variety of tagging options for this task. Infrared (IR) [850nm] is a convenient medium for signaling across distances of up to 25 feet. It is invisible to the human eye, consumes a small amount of power and its transducers are the size of electronic diodes (physically very small). The power requirements for infrequent communication using

only a small Lithium cell (180mAh) can result in an application lifetime as long as one year. In addition, many monochrome CCD cameras are sensitive to the near IR band as well as the visible spectrum, thus providing an inexpensive sensing device that can spatially localize an IR source. In this way physical objects can be augmented with IR-micro-tags that have the desired small size and low power-consumption to make them a viable option. Position sensing can be achieved through image processing by inexpensive and easily deployed CCD cameras. This kind of image analysis is considerably simpler than the generalized case of recognizing arbitrary objects. Our belief is that IR is a pragmatic technology with a fast track for building the required location system.

In more detail, assume the IR microtags periodically beacon their ID (there are however a number of creative ways a tag can be triggered). A CCD camera is used to spatially locate the flash in its image, perhaps requiring reference markers within the same scene to calculate the position. We now need to determine the ID of the tag that has just signaled. There are two strategies for extracting the ID.

In-band: the coded flashing of the IR-tag may be tuned so that the frame rate of the camera is synchronized with its transmission clock. In this case, successive frames of a video stream will result in light and dark patches at the same position in the camera's field of view. A relatively lightweight image processing algorithm can therefore extract the encoded data and generate the ID. The system, although slow for data transfer from a single tag, can process multiple tags without data collisions or contention.

Out-of-band: a coded ID transmission is modulated onto the IR carrier at a high frequency (compared to In-band) using techniques that are well known (e.g., IrDA physical layer). This data can be detected by an independent IR-receiver diode, a pre-amp and a decoder in order to recover the ID. The time the data is received needs to be correlated with a unique flash seen by the camera. If a collision occurs, data will be lost in this system. It is possible to reduce a tag transmission time to a minimum and rely on retransmission, randomization and statistics to successfully read a large number of tags in a suitably long period of time.

Devices that operate as described in 1 & 2 above can be built into a single IR tag that is about the diameter of a 1 cent coin and stands only 5 coins thick. Further miniaturization may be possible.

4.2 Integrating Tags with Applications

By using the tagging technology described above, we may consider each object in a scene, and its position relative to every other object, as part of a unified physical UI. By careful interpretation of an image and by the appropriate assignment of function to objects, a computing system set up to coordinate this environment could then initiate the appropriate set of actions.

In some cases it may not be possible to permanently attach an active tag to an object. For instance, in the case of a slim paper document, for much of its life we may not wish to have an IR microtag stapled to it. However, this document can be printed with a coded label (perhaps invisible to the human eye). By placing a suitably modified version of our tag over the label, on an occasion when coordination with the office environment is required, the tag can read data from the label it is now obscuring and use it to modulate the document identity onto its IR signal.

4.3 User Interface Models

We will need new user interface models for working in smart spaces. The goal is to present an integrated set of devices that work across an interconnected web of information. There are three primary strategies for meeting this goal.

Information Appliances: Each device in the smart space that has a perceptible user interface should be designed as a specific information appliance. Each appliance will have its own capabilities that are made apparent in its affordances. Simple physical affordances include visual display space, mobility and the existence of a writing surface. As physical objects and tools (such as a magnifying glass, a briefcase and a safe) have particular capabilities with respect to physical artifacts, information appliances should have particular capabilities with respect to computational artifacts.

Information Flow: More than ever before, we work in an information-rich environment. Our enhanced spaces will be of limited use if the information manipulated in those spaces is only accessible on one particular device. Similarly, the interconnections between pieces of information need to be maintained as the point of access changes. This model of an evolving web or flow requires new methods for capturing, storing and retrieving information.

Unconventional Media: As we work within a flow of information using a variety of devices, we will

need to turn to underutilized methods for interacting in these spaces. Output that can be processed with only peripheral attention is useful for maintaining awareness of the state and activity in a smart space. Possible media for peripheral output includes non-speech audio, shadows and tactile stimulation. New physical input methods such as tilt, rotation and pressure will aid in making interaction with individual appliances more natural and intuitive.

4.4 Scenario-Based Design and Implementation

As we achieve demonstrable results in each of these three areas, we believe it is important to integrate these advances into office applications and services. In a possible scenario, the contents of a paper document could be "thrown" to an augmented whiteboard for shared review and editing. Likewise, the whiteboard display could augment the use of display-limited or display-less devices such as a PDA or the telephone. For example, the whiteboard could augment handwritten to-do lists with a record of calls to be returned and appointment reminders. By tracking the location of various pieces of paper, the results of physical activities such as sorting, grouping or prioritizing could be recorded. The results could then be accessed on a number of devices including a PDA, an augmented whiteboard or a desktop computer.

5 REFERENCES

1. Communications of the ACM, Special Issue on Augmented Environments, 36 (7), 1993.
2. Ishii, H. and Ullmer, B. "Tangible Bits: Towards Seamless Interfaces between People, Bits, and Atoms," in Proceedings of CHI'97, ACM, March 1997.
(http://tangible.www.media.mit.edu/groups/tangible/papers/Tangible_Bits_html/index.html)
3. Mynatt, E.D., Back, M., Want, R., and Frederick, R. "Audio Aura: Light-Weight Audio Augmented Reality." Published in the Proceedings of the Tenth ACM Symposium on User Interface Software and Technology (UIST), Banff, Alberta, Canada, October, 1997.
(<http://www.parc.xerox.com/mynatt/pubs/audio-aura-uist97.ps.Z>)
4. Want, R., Hopper, A., Falcao, V. and Gibbons, J., The Active Badge Location System, ACM Transactions on Information Systems. Vol. 10 (1), 1992, pp. 91-102.
(<ftp://ftp.ori.co.uk/pub/docs/ORL/tr.92.1.ps.Z>)
5. Want, R et al. "The Parctab Ubiquitous Computing Experiment" Book Chapter #2, Mobile Computing, Kluwer Publishing, Edited by Tomasz Imielinski Chapter 2. pp45-101, ISBN 0-7923-9697-9, February 1997.
(<http://www.ubiq.com/parctab/csl9501-abstract.html>)
6. Weiser, M. and Brown, J.S. (1995) Designing Calm Technology
(<http://www.ubiq.com/weiser/calmtech/calmtech.html>)
7. Weiser, M. "Some Computer Science Problems In Ubiquitous Computing." Communication of the ACM. July 1993.
(<http://www.ubiq.com/hypertext/weiser/UbiCACM.html>)
8. Weiser, M. The Computer of the 21st Century. Scientific American 265(3) 1991, pp. 94-104.

Index to Authors

This page intentionally left blank

Index to Authors

- Almeroth, K., 7-3
- Basu, S., 7-8
Bauer, M., 7-135
Bianchi, M., 7-14
Blumenthal, D., 7-57
Brewer, F., 7-57
Brooks, C., 7-103
Brown, M., 7-121
Brumitt, B., 7-126
Burdea, G., 7-30
- Colbath, S., 7-62
Cong, G., 7-113
Czerwinski, M., 7-126
- De Lucia, D., 7-3
DeFanti, T., 7-121
- Estrin, D., 7-41
- Finn, G., 7-19
Flanagan, J., 3-1, 7-30
Freeston, M., 7-38
- Gemperle, F., 7-76, 7-78, 7-80, 7-135
Govindan, R., 7-41
- Haynes, L., 7-68
Heidemann, J., 7-41
Herman, H., 2-14
Hodes, T., 7-44
Hong, J., 7-82
Hudson, S., 7-52
- Ilgen, M., 7-131
Iltis, R., 7-57
- Jan. E., 7-8
- Kasabach, C., 7-76, 7-78, 7-80, 7-135
Katz, R., 7-44
Khosla, P., 6-1
Krumm, J., 7-126
Kubala, F., 7-62
Kwan, C., 7-68
- Landay, M., 7-82
Lee, H., 7-57
- Lucente, M., 7-8
Luhrs, R., 7-86
- Makhoul, J., 7-62
Marbukh, V., 6-1
Mark, W., 7-98
Marsic, I., 7-30
Martin, R., 7-135
Mazer, M., 5-1, 7-103, 7-105
McRobbie, M., 7-121
Medl, A., 7-30
Meyers, B., 7-126
Mills, K., 2-3
Mizell, D., 7-110
Myers, D., 7-68
Mynatt, E., 7-140
- Neti, C., 7-8
Newman, M., 7-82
- Obrackzka, K., 7-3
- Pacione, C., 7-76, 7-78, 7-80
Parvin, B., 7-113
Paschall, D., 7-117
Pausch, R., 4-1
Popek, G., 7-137
- Ranganathan, M., 5-1, 7-105
Reed, D., 7-121
Reiher, P., 7-137
Ressler, S., 4-1
Robbins, D., 7-126
- Shafer, S., 7-126
Scholtz, J., 2-3
Shynk, J., 7-57
Sieworek, D., 7-76, 7-78, 7-80
- Sirosh, J., 7-131
Sollins, K., 2-27
Stanford, V., 1-1, 3-1
Stevens, R., 7-121
Stivoric, J., 7-76, 7-78, 7-80, 7-135
- Tay, C., 7-113
Taylor, J., 7-113
Touch, J., 7-19
- Tsai, M., 7-137
- Varvarigos, M., 7-57
- Want, R., 7-140
Weiser, M., 7-140
Wilder, J., 7-30
- Xu, R., 7-68
Yarvis, M., 7-137
- Zyda, M., 7-121