Ubiquitous Computing: Trends and History

Lecture 2

Review: What is Ubiquitous Computing?

- Immerses computers in a real environment
- Sensors support interact with and control the environment.
- Limited power supply, storage, memory and bandwidth.
- Operate unattended (much like embedded systems).
- Devices are mobile/wireless.
- May reside on a person (wearable computing).
- Have special peripherals.
- Contrast this with virtual reality which immerses humans in a computer generated artificial environment.

Historical Origins and Trends

Computers are becoming smaller and cheaper over time

- Originally few computers many operators
  - Machines Expensive and Large
  - People (relatively) cheap
- Trend toward more computers per person
  - Users may not be tech savvy
  - Even tech savvy users have limited time
  - Minimal intervention is required

People don't want to be separated from their data

- But spying on users upsets them
- And can violate laws - security is important
- Mobility and wireless access are critical.

Some Popular Views

Many visions were popularized in the press

- First to work on it, although other visionaries preceded him
- Entertainment Industry (Ian Fleming, Gene Rodenberry)
- Vanaver Bush's seminal article [1] As We Might Think predicted the WWW and Ubiquitous Computing in 1945!
- Vernor Vinge (retired Computer Science Professor and Science fiction writer) has interesting ubiquitous computing visions.
- Movies: The Terminator, numerous Philip K. Dick books and screen plays (Blade Runner, Total Recall, Minority Report).

Has been popular in the research community for over a decade

A historical view from 1993

Weiser [4] is credited with popularizing ubiquitous Computing

- Work began at Xerox PARC in 1988
- Ubiquitous Computing is NOT:
  - ▷ virtual reality — real world provides input, not computers!
  - ▷ A PDA or PC — Called an intimate compute, takes your attention to get it to do the work
- Ubiquitous Computing
  - ▷ Supports a world of fully connected devices
  - ▷ Ensures information is accessible everywhere
  - ▷ Provides an intuitive, nonintrusive interface, feels like you are doing it
- Challenges Include:
  - ▷ Wireless bandwidth — high speed and highly multiplexed
  - ▷ Handling mobility
  - ▷ User Interface (window systems)

Computational Issues Back in 1993

Weiser [5] started work in 1988 and reported in 1993

- He didn't want an intimate computer
- Initially Virtual Reality (VR) seemed to have similar design approaches
  - ▷ VR gets the computer out of the way (supports intuitive interaction)
  - ▷ But VR has serious problems
    - ▷ Making sufficiently realistic simulations is expensive (and probably will be for decades)
    - ▷ VR locks users away from reality
- Multimedia is different as it seeks to attract your attention
- Different from Assistants (e.g. PDA or Intelligent Agents) which work for you
  - ▷ Imagine a heavy rock being lifted by an assistant
  - ▷ Imagine being able to lift the rock yourself (effortlessly)
- Informal Goal: Computing for every day life

Weiser's Design Goals

Used the construction of everyday things

Focused on physical affordances

- Wall Sized Interactive Surface
- Notepad
- Tiny computer (e.g. light switch sized)

Developed Hardware Prototypes:

Weiser's Design Approach

Liveboard - digital white-board

Tab - Tiny information portal

- Power is a major issue, cannot always change batteries
- Batteries large and heavy relative to other components
- Used COTS Intel 8051 microcontroller

Pad - Notebook based device

- Originally tethered Sun SBus, later untethered
- Always ran XWindows
- Used Pen interface
- Built in house to satisfy design goals:
  - Control of balance in prioritizing design criteria
  - Ability to ensure inclusion of design features
  - Ease of expansion and modification

Computer Science Time Capsule 1993

Desktop Processor Architecture of the day

- Intel Pentium Released in 1993, 3.1 million transistors.
- Blazing Speeds of 60 and 66 MHz, about 100 Mips
- Memory Speeds were about 66 MHz
- RISC architectures were faster (but were mostly UNIX based).
- Windows 3.1 Popular (some people ran MS DOS still).
- Windows NT was brand spanking new!
- Linux was 2 years old.
- WWW was just beginning to be noticed, internet mostly in labs
- Wireless almost exclusively meant cell phone back then

Weiser's Computational Issues

Computer Science Issues

- Reduce Power Consumption

$$\text{Power} = \text{Gate Capacitance} \times \text{Supply Voltage} \times \text{Clock Frequency} \tag{1}$$

  - ▷ Chips in 1993 didn't have power saver modes
  - ▷ Most chips had failures when underpowered
- Wireless data protocols were not widely deployed, still in the lab
- Pens for very large displays

Weiser's Wireless Networking Issues 1 of 2

Media Access Control (MAC) protocols

- Supports multiplexing broadcast media
- Chose MACA - avoids undetected collisions which garble signals.
  - ▷ MACA uses time division multiplexing
  - ▷ All nodes must have the same transmission radius
  - ▷ Nodes don't transmit when the channel is busy.
  - ▷ Message sizes are advertised (to let listeners know how long they need to wait).
  - ▷ When a node wants to transmit it sends a Request to Send N Bytes (RTS).
  - ▷ When the receiver detects the channel is clear it sends a Clear to Send (CTS) N Bytes
  - ▷ If a collision occurs all stations should back off the same amount.

Physical layer was challenging

- FCC regulations and technology drove them to 900 MHz bandwidth
- 1990 technology was not up to spread spectrum
- But my office phone used to have it (before it failed)
- Went with low power frequency shift keying (FM) approach
- Low power reduces media contention and avoids FCC regulations

Weiser's Wireless Networking Issues 2 of 2

Wide Bandwidth Range

- MACA needed fairness guarantees
- and differentiated QoS
- Added a Not Clear to Send (NCTS) packet for bandwidth reservation by base stations.

Real Time Multimedia Protocols

- QoS needed for streaming multimedia
- May need higher layer

Packet Routing

- Need base station load balancing
- IP not designed to support mobility
  - However, it is dominant
  - OSI ISO 8473 Connectionless Network Protocl (CLNP) has some mobility support, but is less popular

Weiser's Interaction Substrates

Interaction Substrate is what we call the UI Toolkit

- Windowed Mouse Point and click (WiMP) are still dominant
- XWindows designed for networked use
  - Specifies policy not appearance
  - Attempts to be device independent (e.g. units of length measures used are not)
- Display areas vary between physical devices
  - Pads often have tiny interaction areas
  - Liveboards have huge interaction areas
- Input devices depend on size
  - Pads need pens, since keyboards are too big.
  - Pens needed special script since general handwriting mechanism is too hard
- Added support for migrating windows in X.
- Proposed support for low bandwidth network connections (vary protocol according to bandwidth).

Weiser's Applications

Applications

- Locating People
  - ▷ Data acquired from:
    - ▷ Log ins to workstations/terminals
    - ▷ An Active badge system (smart badges?)
  - ▷ Useful for
    - ▷ Automatic call forwarding
    - ▷ Shared Drawing Tools
  - ▷ An Active badge system (smart badges?)
- Shared Drawing
  - ▷ Data Representation
    - ▷ Object (vector) based
    - ▷ Bit mapped
  - ▷ UI Issues
    - ▷ How to handle multiple cursors?
    - ▷ Use gestures or not?
    - ▷ Use an ink based or character recognition model of pen input?

Impending Application Concerns

Characteristics of future Ubicomp Applications

- Smart environment (hiding computing in walls/infrastructure)
- Virtual Communities
- Information filtering (streaming data management)

Weiser expects security concerns

- Preserve privacy by aggregating information
- Nontechnical issues are important

Computational Issues raised by Weiser

Cache Coherence Problem

- Classical distributed computing problem
- Consider multiprocessor machine with a single address space
- If 2 processors have the same location cached, how do they make sure they see the same value?

How close to the theoretical optimum can on-line cache coherence algorithms get in practice?

Especially if pages can be compressed.

Mann's Definition of Wearable Computing (1998)

Steve Mann [2] states a wearable computer is:

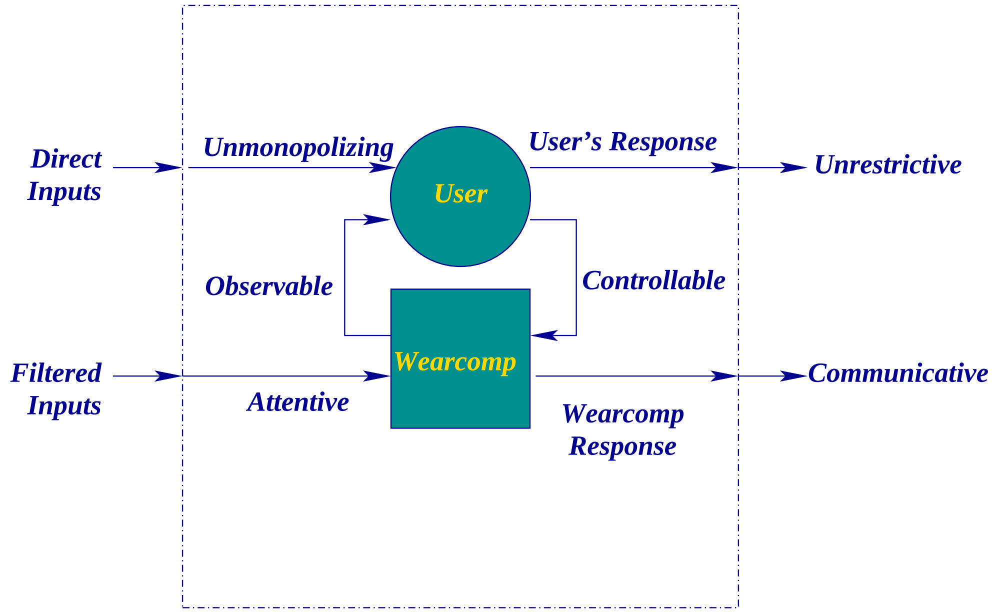
- Subsumed into the personal space of the user
- Controlled by the user and
- Always on and always accessible.

Modes of Operation (how does interaction work?)

- Constancy: The computer runs continuously, and is "always ready"
- Augmentation: The computer helps the user to do other stuff by enhancing his mind or senses
- Mediation: The computer filters information relayed to the user and regulates what information the user wishes to disclose

The Six Attributes of Wearcomp

- Unmonopolizing of the user's attention.
- Unrestrictive to the user: ambulatory, mobile, roving,
- Observable by the user, can alert you when necessary.
- Controllable by the user: responsive.
- Attentive to the environment: Environmentally aware.
- Communicative to others.



Aspects of wearable computing and personal empowerment

- Photographic memory: Perfect recall of collected information.
- Shared memory: Individuals may share their recorded experiences.
- Connected collective humanistic intelligence, facilitate collaboration
- Personal safety: The wearcomp can allow for distributed protection from danger.
- Tetherless operation: Wearable computing affords and requires mobility.

Satyanarayanan's Approach (2001)

Satyanarayanan [3] (Satya for short) discussed current issues:

- Calls Ubicomp Pervasive Computing
- Several Example Groups:
  - ▷ Project Aura at CMU
  - ▷ Edeavour at UBC
  - ▷ Industrial AT&T research Cambridge U.K. (Stajano?)
  - ▷ IBM TJ Watson (Westchester County, NY)
- Contrasts with Prior Art/Related Fields
  - ▷ Distributed Systems (tethered)
  - ▷ Mobile Computing (untethered)

Satya characterizes distributed systems as having (1980's research):

- Remote communication — protocol layering (e.g. rpc's, timeouts, 2 phase commit).
- Fault Tolerance - Atomic/nested/distributed transactions, 2 phase commit.
- High Availability — Optimistic/Pessimistic replica control, mirrored execution and Optimistic recovery
- Remote Information Access - Caching, Code Migration, distributed file systems and distributed databases.
- Security - Encryption for mutual authentication and privacy.

Mobile Computing (1990's research) adds :

- Mobile Networking - Mobile IP, Ad Hoc protocols, Wireless TCP
- Mobile Information Access - disconnected operation, bandwidth adaptive file access, selective control for data consistency.
- Support for adaptive applications - Adaptive Resource Management, Transcoding by Proxies
- System Energy Management - Energy aware adaptation, Architectural Support
- Location Sensitivity - Location sensing, and location aware system behavior.

How is Pervasive Computing Different?

Smart Spaces

- Use Computing Infrastructure embedded in a building to assist the user.

Invisibility

- The computer should not distract the user

Localized Scalability

- Adding Ubicomps to a smart space should not overtax the infrastructure

Masking Uneven Conditioning

- In spite of variable smart space deployments, a user should have a consistent experience

Can We Improve Pervasive Computing

Proactive handling of user needs

- Ability to predict system behavior given conditions e.g. Wireless congestion is a low level
- Recognize constraints - want to send e-mail before departing flight
  - ▷ Wait for slow e-mail could cause missed flight
  - ▷ Leave for flight prevents e-mail
  - ▷ Realize that constraint is spatially localized
- Clever use of smart spaces may find alternatives
  - ▷ e.g. Suggest uncongested regions of airport

Self-Tuning - adjust behavior to circumstances

- Sense user intent, predict likely user needs
- Code and Data Migration
  - ▷ Put the access where the user is
  - ▷ Alert smart space infrastructure to prepare for user's arrival
- Alert user to potential constraint violations
  - ▷ Warn user before transmitting confidential data

Drilling Down

Ubicomp provides sort of virtual immersion

- Like VR
- But it goes with the user

Ubicomp devices worn by the user are called clients.

- Not in the client-server sense

We need a layer above the applications (Prism)

- To coordinate the constraints of applications
- To sense user intent

Remote execution support via Spectra

Nomadic file access via Coda

Resource monitoring and adaptation using Odyssey/Chroma

Linux Kernel

Intelligent Networking

User Intent

Guessing User intent is hard

- Hence modern systems don't do this well!
- Why? Generic applications lack enough information
  - ▷ e.g. Viewing Streaming Video and network bandwidth suddenly drops
  - ▷ Should the application:
    - ▷ Wait for less contention
    - ▷ Reduce display fidelity
    - ▷ Tell the user that the service is unavailable
- Bad User Intent systems are intrusive
  - ▷ Do you really want Microsoft's "Clippy" to help?
- Research Opportunities!
  - ▷ Can user intent be inferred, or does the user need to explicitly signal intent?
  - ▷ How can user intent be represented?
  - ▷ How can we measure accuracy in measuring user intent?
  - ▷ Will the attempt to obtain intent cause a burden to the user exceeding the benefit?

Cyber Foraging does nomadic resource discovery

- Cheap computing means "waste" is not so bad
- Is Communication or Computation cheaper?
  - Computation Cheaper — Owner Computes
  - Communication Cheaper — Find a surrogate
  - How to decide? Must know bandwidth, size of inputs, outputs and code to migrate to be sure.
- More Research Opportunities!
  - How can a device best find surrogates?
  - How can trust be established with surrogates?
  - How is load balancing done between surrogates?
  - How much advance notice does the surrogate need to avoid excessive delay?
  - What are the implications for scalability?
  - What system support is needed to make surrogate use minimally intrusive?

Additional Research Areas

Adaptation Strategies — Adjust to variable constraints

- Is reservation based QoS approach correct?
- Is it feasible to use corrective actions to support adaptation?

High Level Energy Management

- Can user intent generate meaningful hints for energy management?
- Can smart spaces and surrogates reduce demand on a mobile device?

Client Thickness — Trade-off between functionality and complexity

Context Awareness — Needed for Minimizing Intrusiveness

- Context is users state and his surroundings - how to represent this?
- What are the merits of different location sensing technologies?

Balancing Proactivity and Transparency

Privacy and Trust

Impact on Layering

Brief Note on Timestamp Ordered Protocols and PDES

In lecture I mentioned briefly about Parallel Discrete Event Simulation (PDES).

Each Simulation Entity has a discrete state

Entities represented via logical processors (LP)

LPs communicate via time stamped messages

LPs advance simulation state (and time) by processing messages.

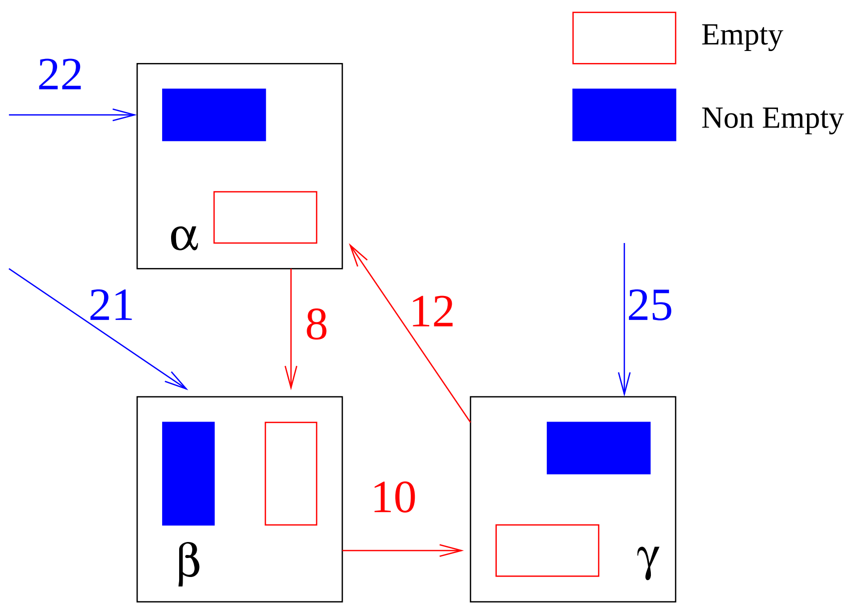
Why is PDES Hard?

- Local Causality Constraint - Must ensure that each LP processes (interfering) messages in nondecreasing time stamp order.
- Some processors may be slower, and late messages (stragglers) are a problem.
- For efficiency, we don't want to restrict order of processing.

Flavors of protocols

- Optimistic - Uses Speculative Execution, with rollback or reverse computation.
- Conservative - Only processes messages when it is safe.

An Example of Deadlock



Conservative Protocols are Susceptible to deadlock

Optimistic protocols not much easier

- Tend to have cascading rollbacks
- Tend to use a lot of memory for checkpoints
- Need to compute Global Virtual Time (time of last correctly processed event).
- Hard to know when checkpointed data is safe to discard

References

[1] Vanaver Bush. As we may think. The Atlantic Monthly, July 1945. On line at http://www.csi.uottawa.ca/ dduchier/misc/vbush/awmt.html.

[2] S. Mann. Definition of "wearable computer". On line at http://wearcomp.org/wearcompdef.html, 1998. From Mann's Keynote Address entitled "WEARABLE COMPUTING as means for PERSONAL EMPOWERMENT" presented at the 1998 International Conference on Wearable Computing ICWC-98, Fairfax VA, May 1998.

[3] M. Satyanarayanan. Pervasive computing: Vision and challenges. IEEE Personal Communications, pages 10–17, August 2001.

[4] Mark Weiser. Hot topics: Ubiquitous computing. IEEE Computer, October 1993. On line at http://www.ubiq.com/hypertext/weiser/UbiCompHotTopics.html.

[5] Mark Weiser. Some computer science issues in ubiquitous computing. CACM, 36(7):74–83, July 1993.