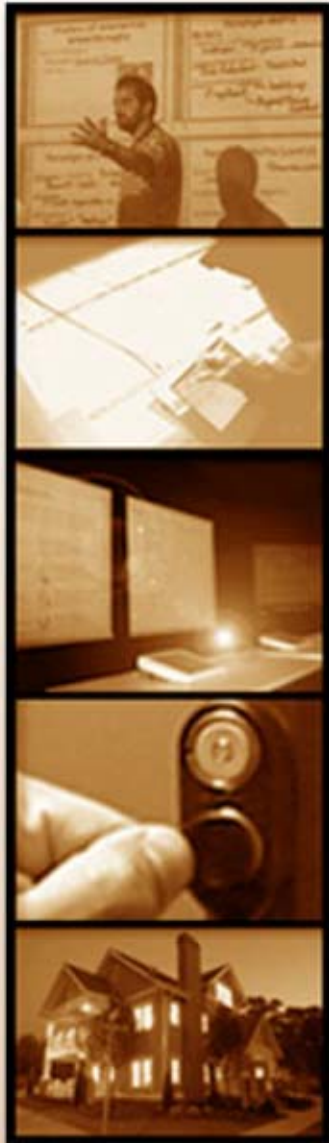


# UbiComp



## Supporting the Construction of Context-Aware Applications

Anind K. Dey

Intel Berkeley Lab

Intel Research

Future Computing  
Environments



Georgia Institute of  
Tech  
Technology

# How UbiComp is Different from Traditional GUI

- Not only direct interaction for input and output
- Need to use RANGE of explicit AND implicit interaction
- Different computing paradigm

# Context and Context-Awareness

- Focused on input
- Context: *any information relevant to an interaction that can be used to characterize the situation of an entity*
- Context-awareness
  - General model of interactive computing
  - Addresses subset of ubicomp problems: input

# Value of Context

- Potential for improved usability
  - Very important for mobile users with poor input devices
- "Smarter" applications
- Increased communications bandwidth

# Design Space for Context-Aware Applications

- Toolkit allows exploration of design space
- Basic types of context:
  - **Location**, identity, time, activity
  - **Simple**/singular → complex/multiple
  - Combinations
- Uses of context:
  - **Present to user**
  - Automatically perform set of **services**
  - **Tag** captured information to ease retrieval

# Example

- Tour guides, travel assistants, personalization software
- Reminder to buy milk
  - When to deliver: not time/location specific
  - How to deliver: appropriate modality

# Outline

- Motivation
- Problems dealing with context
- Contribution: Context Toolkit
- Validation:
  - Design space and applications
  - Building more realistic applications
- Conclusions and future work

# Building Applications

- M. Weiser: The whole point of ubiquitous computing, of course, is the applications.



# Building Applications

- M. Weiser: The whole point of ubiquitous computing, of course, is the applications.
- But ... what if the applications are hard to build? And, what if this inhibits our ability to build compelling applications?

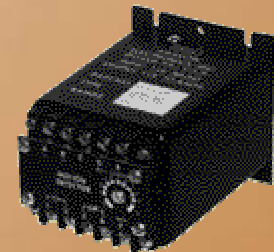
# Issues in Context-Awareness

- What is context?
  - Representation of context
  - Application domains
  - Which behaviors to support
  - When to execute behaviors
  - Privacy, Quality of Service, ...
  - Evaluation of applications
- 
- make it easier to build ►► explore



# Why Context is Hard to Use

- Acquired from sensors
  - Not just keyboards and mice - lots of heterogeneous devices
- Need to abstract data
- Distributed
- Dynamic

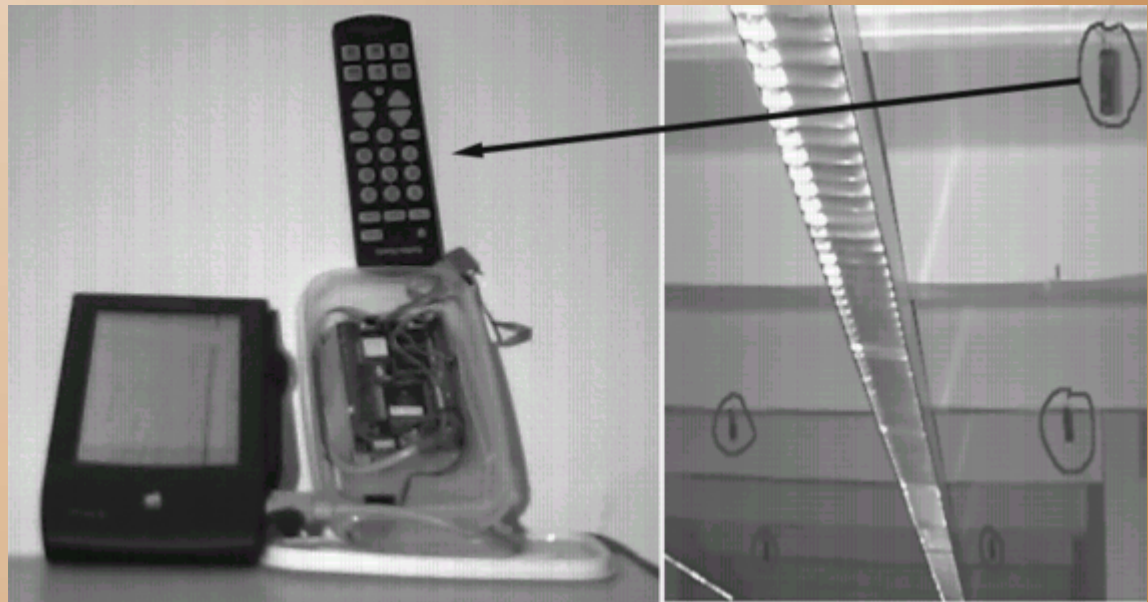
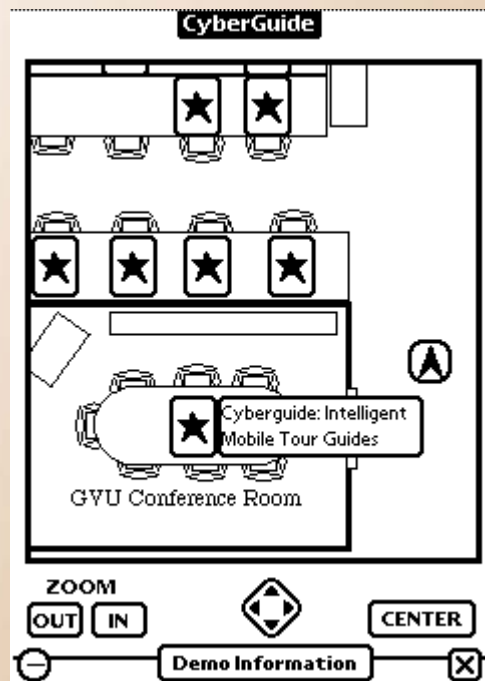


# Results of Difficulties

- *Ad hoc* application building
  - Difficult to build, reuse and evolve
- Small variety of sensors
- Small variety of context: mostly *location*
- Few applications, mostly simple: mostly *presenting context*
  
- Practical: difficult to prototype, test and evaluate

# Why Applications are Hard to Build: A Case Study

- Cyberguide case study: no separation of concerns



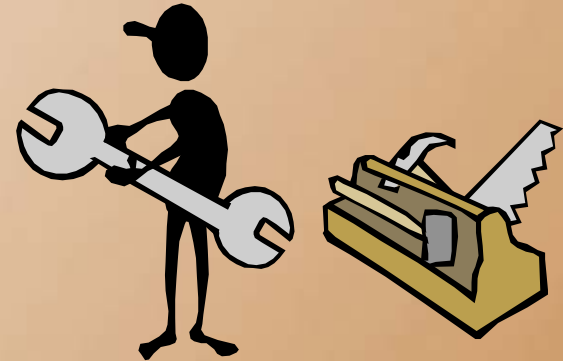
# Outline

- Motivation
- Problems dealing with context
- **Contribution: Context Toolkit**
- Validation:
  - Design space and applications
  - Building more realistic applications
- Conclusions and future work

# Need Programming Support

- **Goal: make application development easier**
- Identified number of requirements for architectural support and design process
- Examined existing support and determined how developers might think about building context-aware applications
- Developed Context Toolkit: architecture with supporting library of components

# Related Work



- Existing Context-Aware Systems
  - Schilit (Columbia, 1995)
  - Stick-e notes (Pascoe, Kent, 1996)
  - CyberDesk (Dey, Georgia Tech, 1997)
  - CALAIS (Ward, Cambridge, 1998)
  - MUSE (Castro, UCLA, 2000)
  - Context-Awareness SDK (Tangis Corp., 2000)
- Proposed/Related Systems
  - Situated Computing Service (HP, 1997)
  - Contextual Information Service (Pascoe, Kent, 1998)
  - HIVE (Minar, MIT, 1998)
  - OAA (Cohen, OGI, 1996)




# Research Contributions

- Conceptual framework requirements
  - Provide framework for designing apps more easily
  - Lower threshold to enable more designers
- Context Toolkit
  - Implementation and exploration of design space
- Support investigation of complex problems and more realistic apps
  - Raise ceiling
  - Privacy, uncertainty, security, end-user programming

# Toolkit Requirements

- Context specification
- Discovery
- Separation of concerns
- Storage
- Constant availability
- Transparent communications
- Interpretation

# Design Process

1. Specification
2. Acquisition
3. Delivery 
4. Reception
5. Action

# Design Process

1. Specification

2. Acquisition

3. Delivery 

4. Reception

5. Action

1. Specification

2. *Acquisition*

3. Action

# Time for a Big Insight!

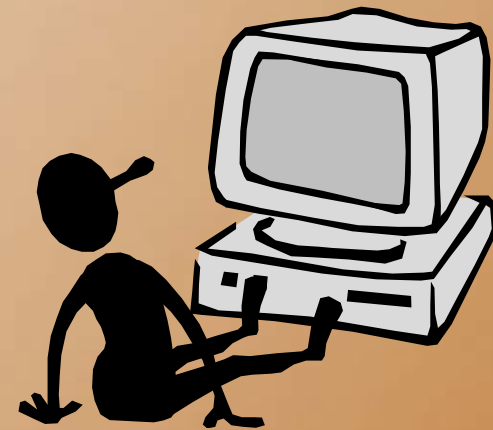
- Have a design process - complex and simplified
- Have a set of architecture requirements
- Need to figure out how to support these



# Look to input handling

- Graphical User Interface (GUI) widgets

- separation of concerns
- callbacks and attributes
- query/subscribe
- common interface

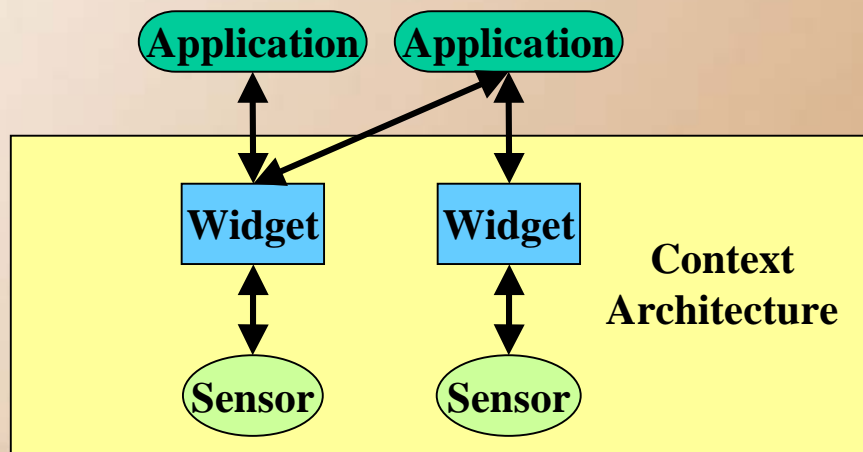


- e.g. button



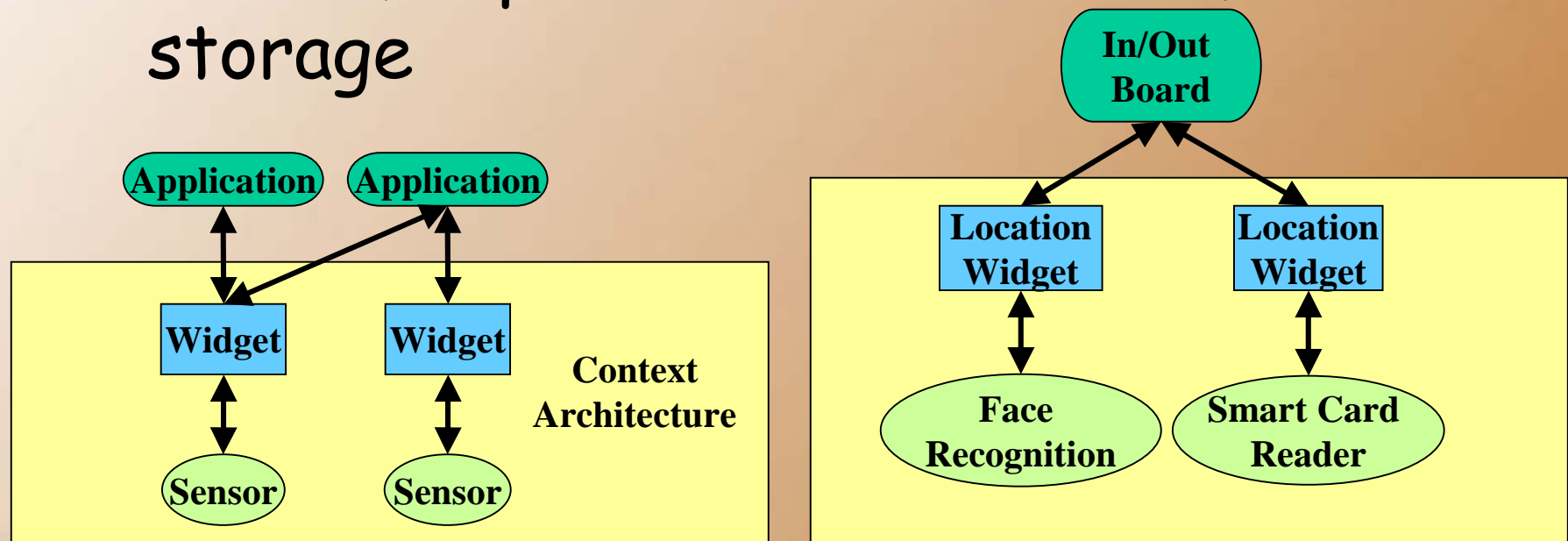
# Context Widgets

- Responsible for acquiring and abstracting data from particular sensor, separation of concerns, storage



# Context Widgets

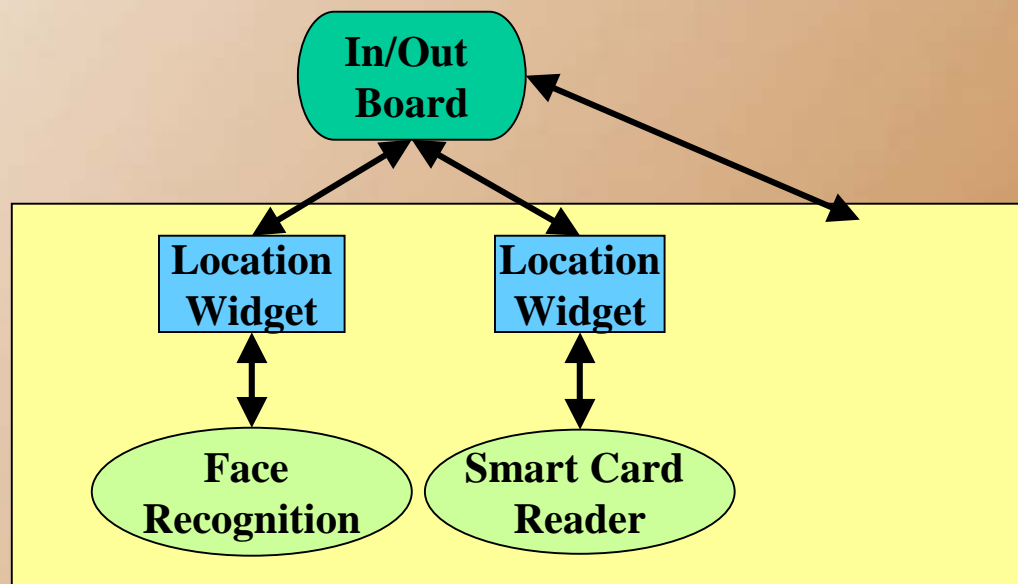
- Responsible for acquiring and abstracting data from particular sensor, separation of concerns, storage





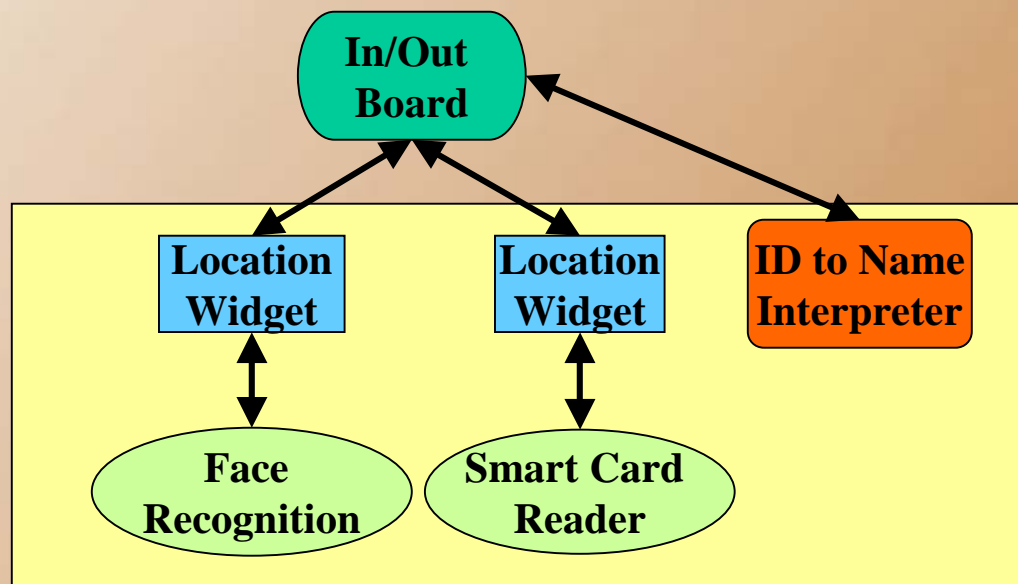
# Context Interpreters

- Convert or interpret context to higher level information
- Context not available at appropriate level



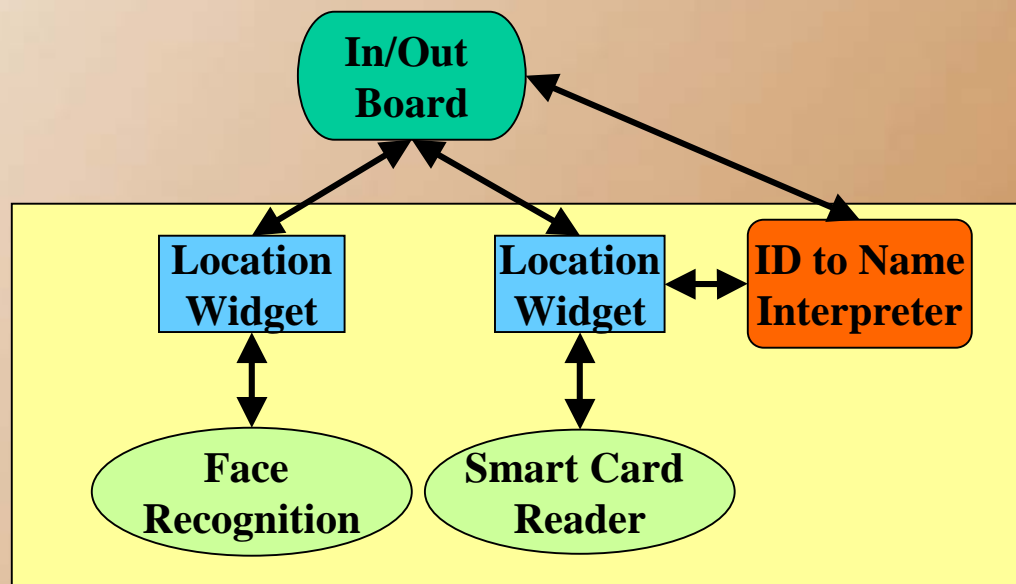
# Context Interpreters

- Convert or interpret context to higher level information
- Context not available at appropriate level



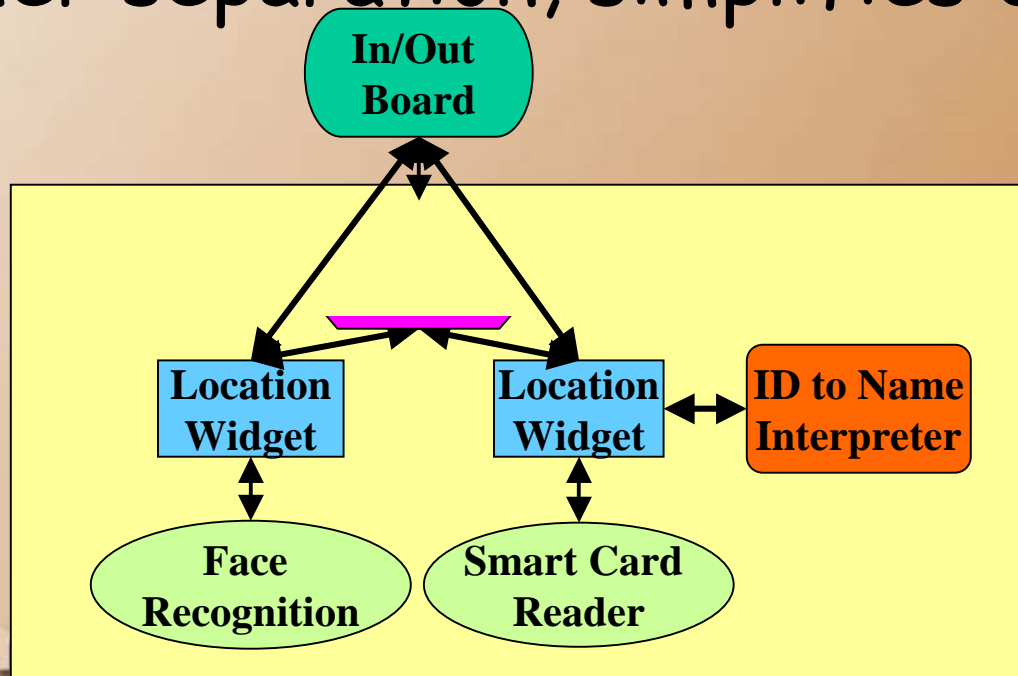
# Context Interpreters

- Convert or interpret context to higher level information
- Context not available at appropriate level



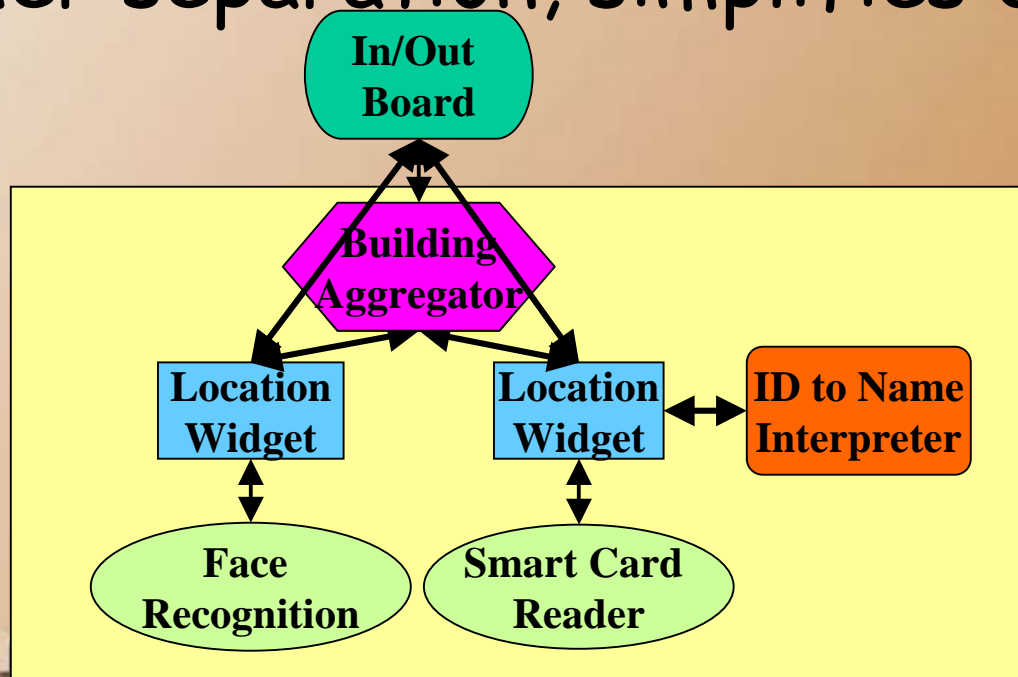
# Context Aggregators

- Collect context relevant to particular entities (recall definition)
- Further separation, simplifies design



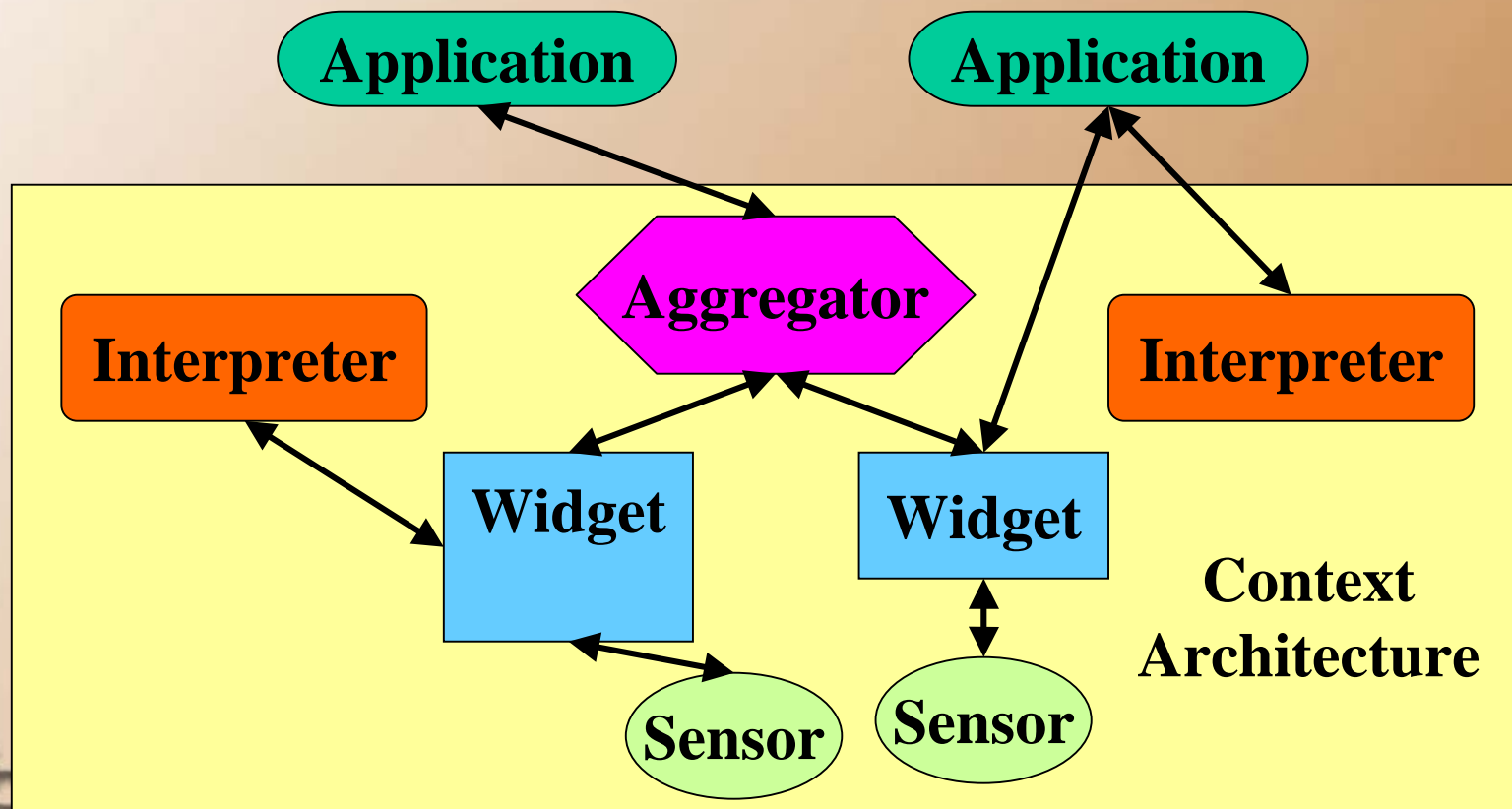
# Context Aggregators

- Collect context relevant to particular entities (recall definition)
- Further separation, simplifies design



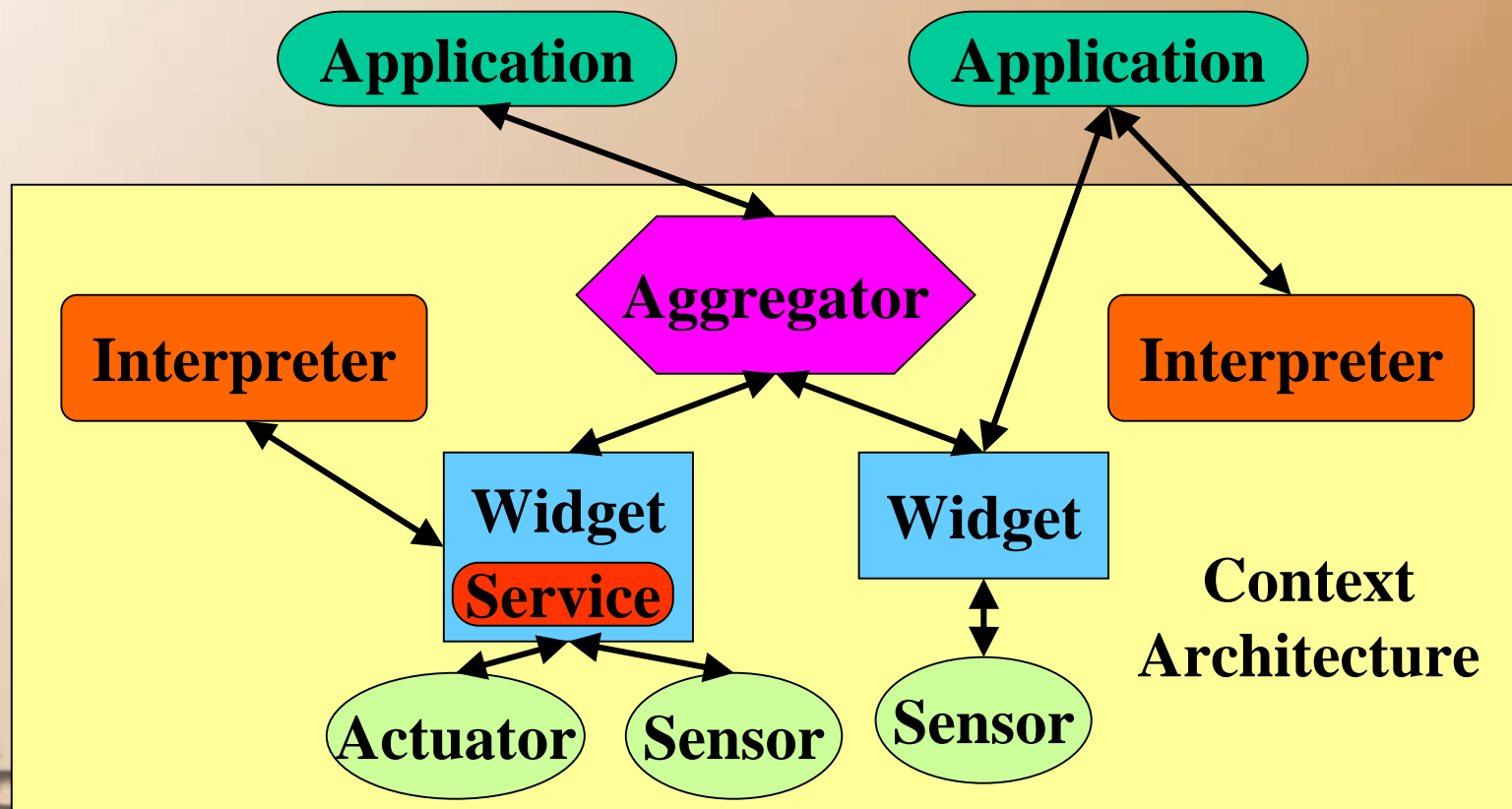
# Context Toolkit Framework

- Supports real-world model/methodology and provides library (distributed: XML/HTTP, input-focused)
- Component model: facilitates building of applications



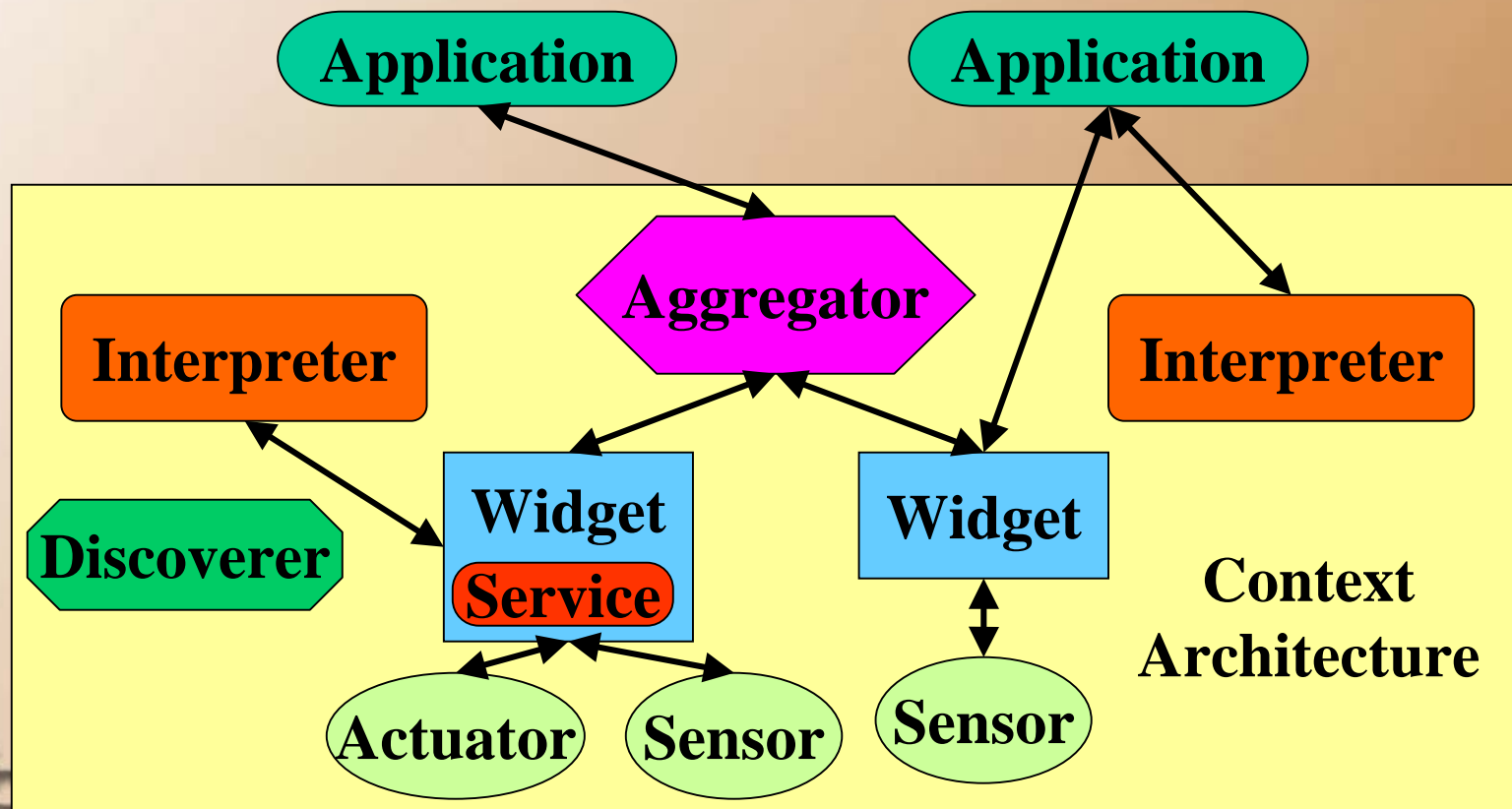
# Context Toolkit Framework

- Supports real-world model/methodology and provides library (distributed: XML/HTTP, input-focused)
- Component model: facilitates building of applications



# Context Toolkit Framework

- Supports real-world model/methodology and provides library (distributed: XML/HTTP, input-focused)
- Component model: facilitates building of applications





# Experiences: Benefits



- Provides separation of concerns
- Lightweight integration and re-use of components
- Easy to create and evolve apps, allowing exploration of the design space
  - Add context to context-less apps
  - Add more context to context-aware apps

# Outline

- Motivation
- Problems dealing with context
- Contribution: Context Toolkit
- **Validation:**
  - **Design space and applications**
  - Building more realistic applications
- Conclusions and future work

# Validation

- Used to build existing applications
- Used to explore the design space
- Used to build more complex and realistic applications

# Additional Validation

- Facilitating larger community outside of Georgia Tech, including:
  - Arch:
    - CMU (mobile agents)
    - Motorola (arch/mobile user apps)
    - Autonama de Madrid (arch/smart spaces apps)
    - British Telecom Labs
    - MIT
    - Trinity College
    - PLAY Research Group
  - Apps
    - Keele (desktop apps)
    - Novator (apps for mobile workers)
    - Technical University Munich/CMU (informal meeting support)
    - Stuttgart
    - SICS, Sweden
    - ETH
    - Philips
    - Telenor
    - Nokia

# Aware Home (MANSE '99)

- Great testbed for context-aware computing
- 3 goals: elderly, infants, everyone
- Context Toolkit is the s/w infrastructure in the Aware Home

# Aware Home (MANSE '99)

- Gr  
col
- 3 g
- Co  
inf



# Design Space for Context-Aware Applications

- Toolkit allows exploration of design space
- Types of context:
  - **Location**, identity, time, activity
  - **Simple**/singular → complex/multiple
  - Combinations
- Uses of context:
  - **Present to user**
  - Automatically perform set of **services**
  - **Tag** captured information to ease retrieval

# Applications Built

- Simple use of location:
  - Turn lights on and off (perform service)
- Location and id (perform service)
  - Information Guide: present info about user's group (CHI '99)
  - Context-Aware Mailing List



# In/Out Board - 3 versions

(CHI '99)

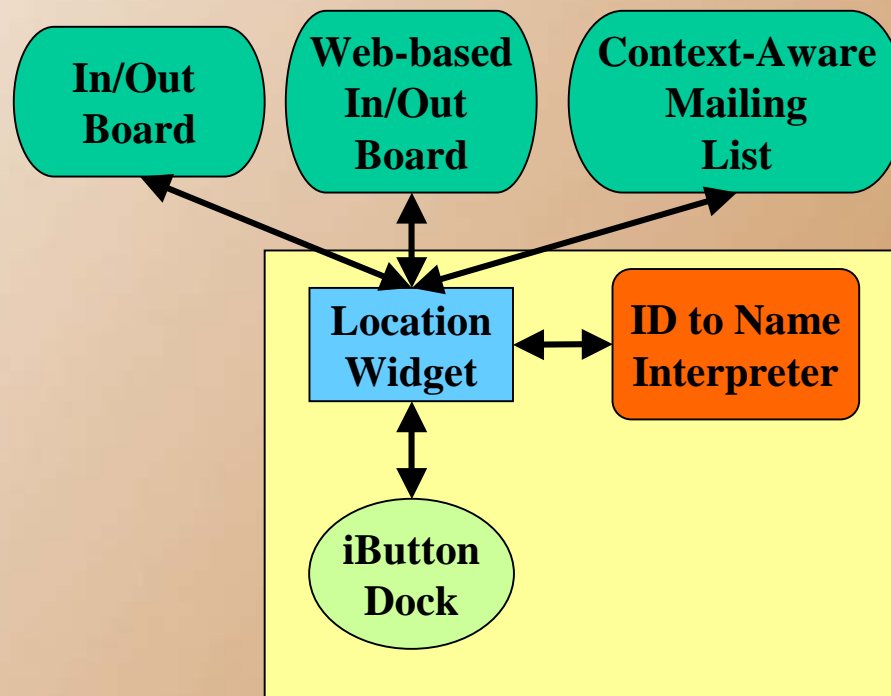
- Context used: location, identity, time
- How used: present context

<b>Gregory Abowd</b> <span style="color:red">●</span> Out 10:50am	<b>Jen Mankoff</b> <span style="color:green">●</span> In 12:08pm
<b>Jason Brotherton</b> <span style="color:green">●</span> In 9:20am	<b>David Nguyen</b> <span style="color:green">●</span> In 11:09am
<b>Anind Dey</b> <span style="color:green">●</span> In 12:08pm	<b>Rob Orr</b> <span style="color:red">●</span> Out 1:25pm
<b>M. Futakawa</b> <span style="color:green">●</span> In 12:00pm	<b>Maria Pimentel</b> <span style="color:red">●</span> Out 5:54pm
<b>Y. Ishiguro</b> <span style="color:red">●</span> Out 10:52am	<b>Daniel Salber</b> <span style="color:green">●</span> In 10:14am
<b>Rob Kooper</b> <span style="color:red">●</span> Out 5:26pm	<b>Brad Singletary</b> <span style="color:red">●</span> Out 2:59pm
<b>Kent Lyons</b> <span style="color:red">●</span> Out 12:27pm	<b>Khai Truong</b> <span style="color:red">●</span> Out 1:25pm

<b>Gregory Abowd</b> in
<b>Jason Brotherton</b> out
<b>Anind Dey</b> in
<b>Tanisha Hall</b> out
<b>Cory Kidd</b> out
<b>Kent Lyons</b> in
<b>Jen Mankoff</b> in
<b>Todd Miller</b> out
<b>Kris Nagel</b> in
<b>David Nguyen</b> out
<b>Rob Orr</b> in
<b>Daniel Salber</b> out
<b>Chris Shaw</b> out
<b>Brad Singletary</b> in
<b>Khai Truong</b> out

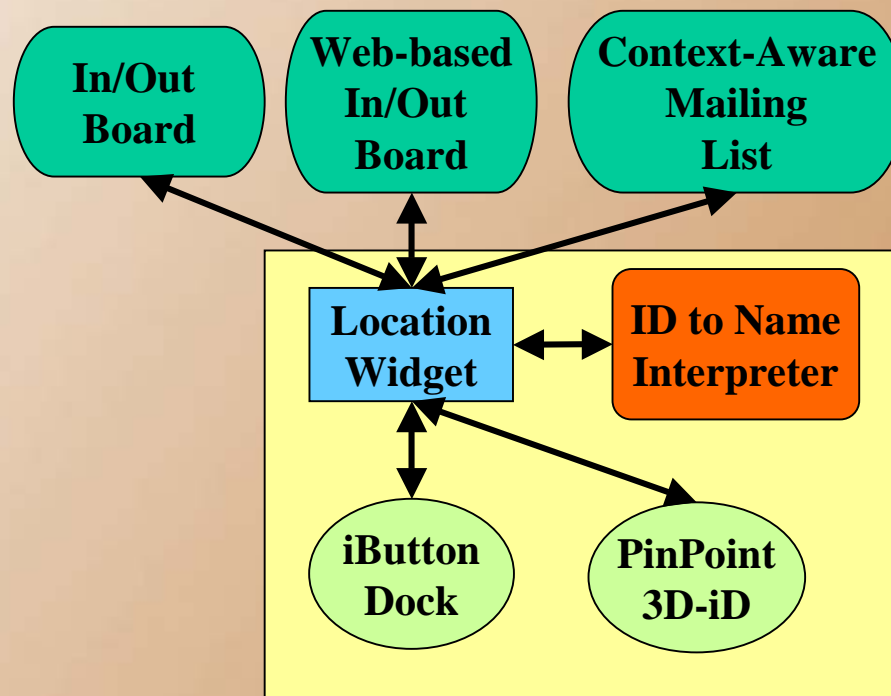
# In/Out Board Architecture

- Simple apps demonstrates support for **reusability** (don't have to re-build infrastructure on per-application basis) and **evolving** applications



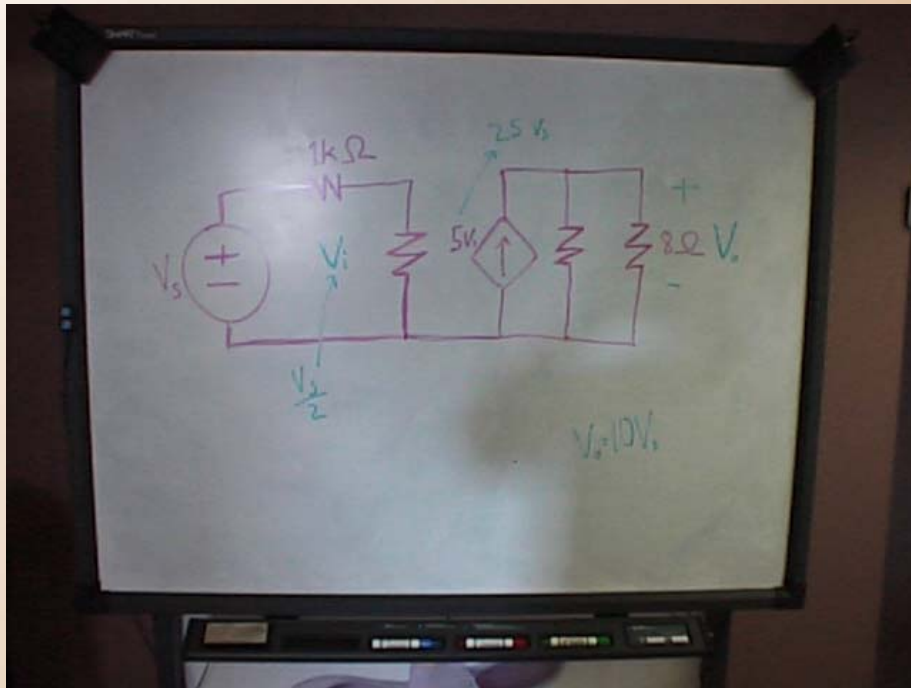
# In/Out Board Architecture

- Simple apps demonstrates support for **reusability** (don't have to re-build infrastructure on per-application basis) and **evolving** applications



# Serendipitous Meetings

- Context used: location, id, time, activity
- How used: present, perform service, tag
- Work done by others in research group



# Serendipitous Meetings

The screenshot shows a window titled "DumboAccess" containing a whiteboard with handwritten notes and diagrams. The notes include "CW", "admin", "Sensor", "GUI", "CA", "Speech/Audio", and "People plus things". A diagram shows a box labeled "W" connected to "Sensor" and "GUI". Another diagram shows a box labeled "CA" connected to "Speech/Audio" and "People plus things". A third diagram shows a box labeled "Person widget" connected to "Person" and "Button".

At the bottom of the window is a playback interface with "Play" and "Stop" buttons, a progress bar, and a calendar grid. The calendar grid shows a selected session (day 4) and a selected day (day 25). Below the calendar are filters for "When: Aug 1998", "Who: Jason Brotherton", and "Where: Mobile SMART Board #1".

Ink written *before* current time is in original color

Ink written *after* current time is in original color

Current time within session

Selected session

Selected day

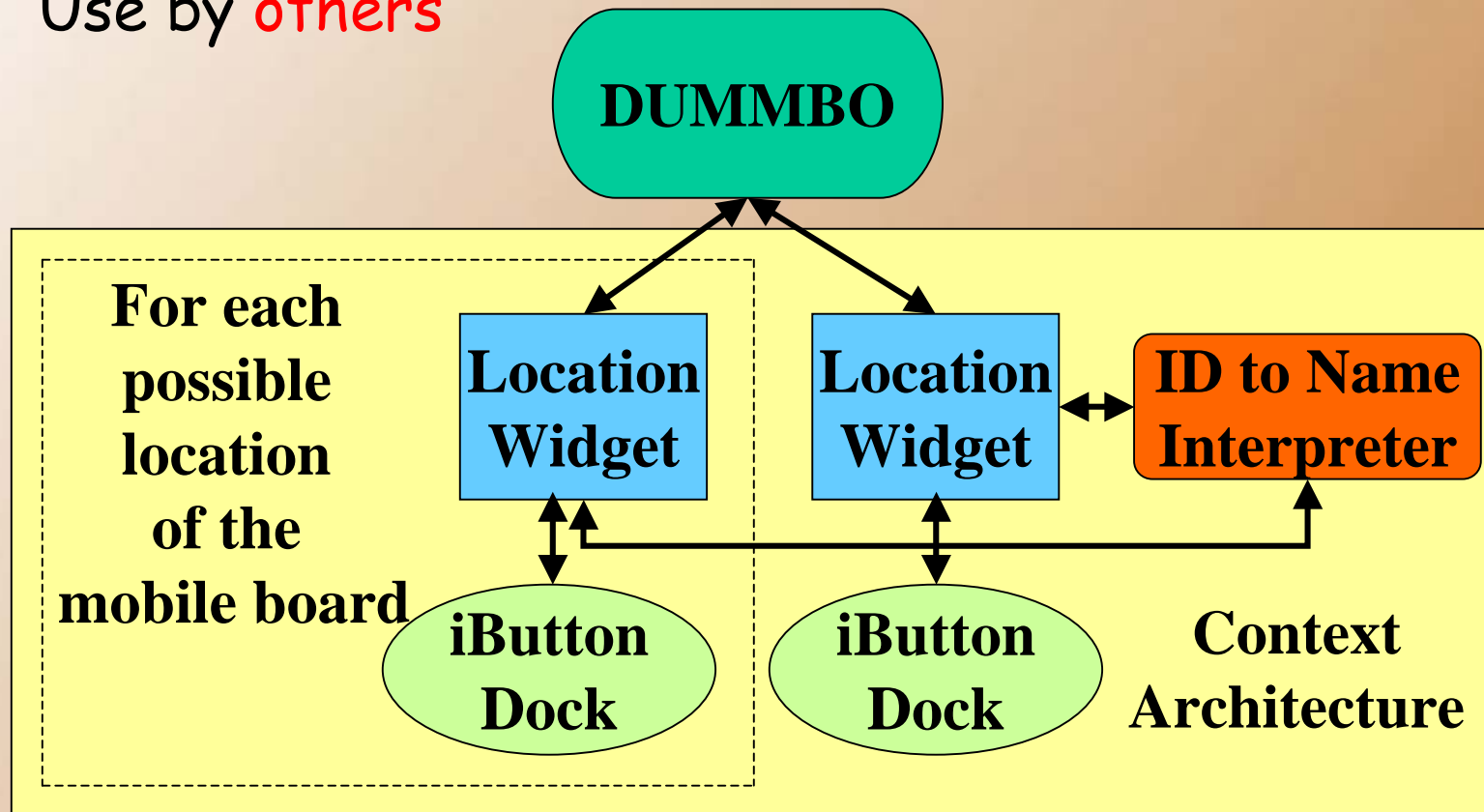
Day containing whiteboard activity

Filters

Playback controls

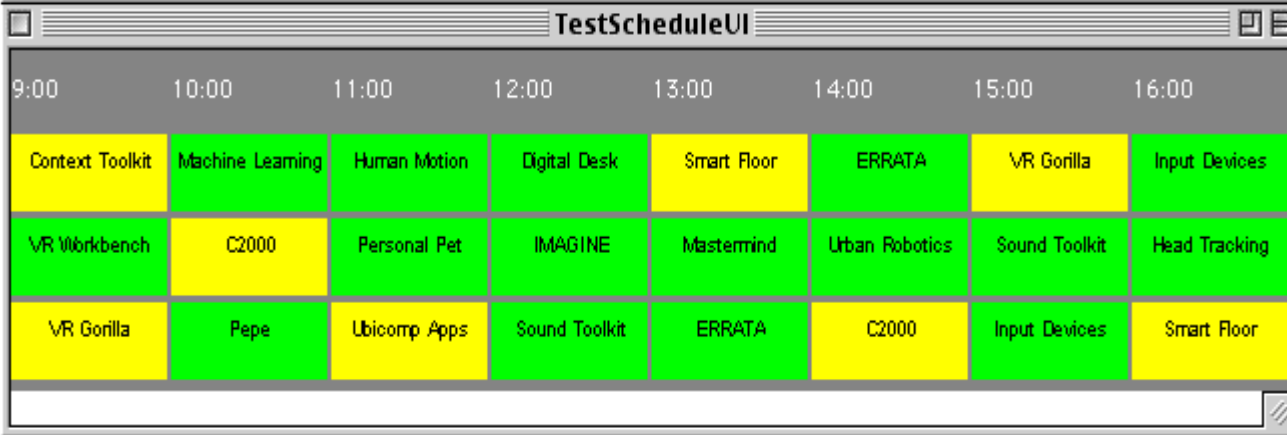
# Meeting Architecture

- Demonstrates support for **evolution**
- Use by **others**



# Conference Assistant (ISWC '99)

- Context used: location, multiple levels of identity, activity, time
- How used: present, service, tag

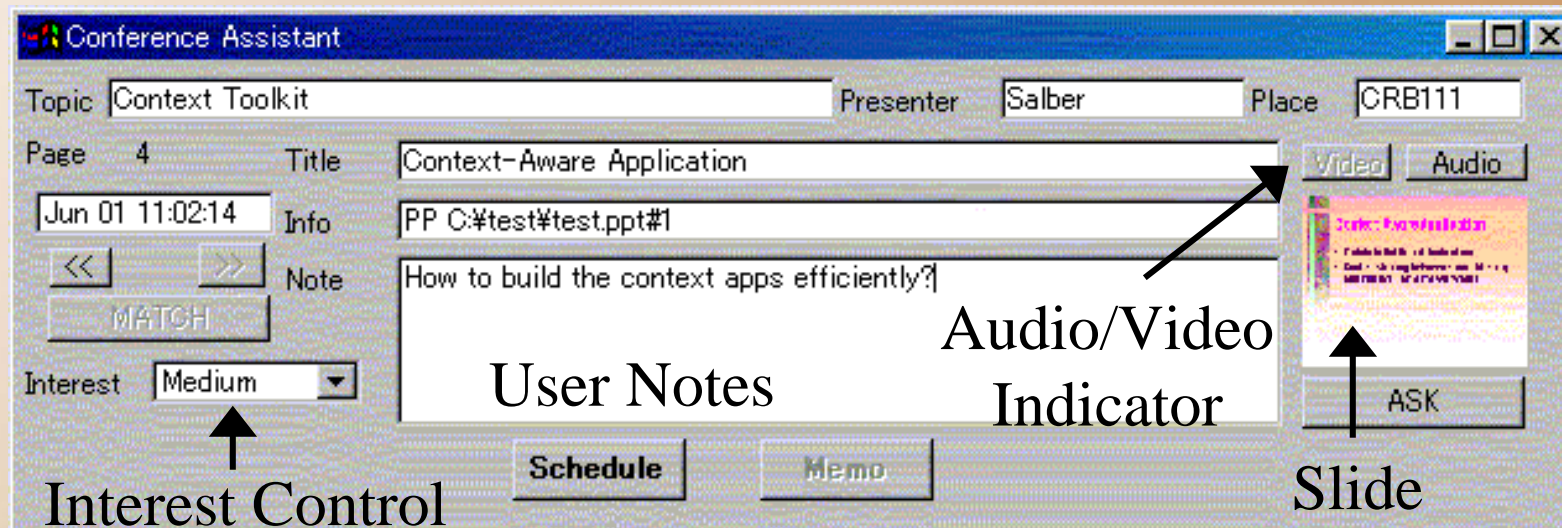


The screenshot shows a window titled "TestScheduleUI" with a grid of colored cells representing a conference schedule. The columns are labeled with times from 9:00 to 16:00. The rows represent different sessions or activities. The cells are colored yellow or green.

9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00
Context Toolkit	Machine Learning	Human Motion	Digital Desk	Smart Floor	ERRATA	VR Gorilla	Input Devices
VR Workbench	C2000	Personal Pet	IMAGINE	Mastermind	Urban Robotics	Sound Toolkit	Head Tracking
VR Gorilla	Pepe	UbiComp Apps	Sound Toolkit	ERRATA	C2000	Input Devices	Smart Floor

# Conference Assistant (ISWC '99)

- Context used: location, multiple levels of identity, activity, time
- How used: present, service, tag





# Conference Assistant (ISWC '99)

- Context used: location, multiple levels of identity, activity, time
- How used: present, service, tag



The screenshot shows a conference assistant interface with a grid of sessions. The grid has 8 columns representing time slots from 9:00 to 16:00 and 3 rows representing different sessions. Each cell contains a session name and a speaker name with a small icon. The background of the grid is dark gray, and the session names are in yellow and green boxes.

9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00
Context Toolkit	Machine Learning	Human Motion	Digital Desk Daniel ▲	Smart Floor	ERRATA	VR Gorilla	Input Devices
VR Workbench	C2000	Personal Pet	IMAGINE Anind ▼	Mastermind	Urban Robotics	Sound Toolkit	Head Tracking
VR Gorilla	Pepe	UbiComp Apps	Sound Toolkit Gregory ~	ERRATA	C2000	Input Devices	Smart Floor

**Conference Retrieval**

Time	Event	Time	Event	Time	Event
9:00	<b>A Daniel Salber-Context Toolkit</b>	9:00	<b>Bill Ribarsky-VR Workbench</b>	9:00	<b>Don Allison-VR Gorilla</b>
9:15		9:15		9:15	
9:30		9:30		9:30	
9:45		9:45		9:45	
10:00	<b>Chris Atkeson-Machine Learning</b>	10:00	<b>A Maria Pimentel-C2000</b>	10:00	<b>Ashwin Ram-Pepe</b>
10:15		10:15		10:15	
10:30		10:30		10:30	
10:45		10:45		10:45	
11:00	<b>Jessica Hodgins-Human Motion</b>	11:00	<b>Ashwin Ram-Personal Pet</b>	11:00	<b>A Anind Dey-Ubicomp Apps</b>

Personal Events:  arrival  departure  question

Person:  Keyword:

**Anind Dey -- Ubicomp Apps**

Schedule

Retrieved slide

Slide text

What Is Context?

- The missing piece
- Information sensed
- Identity, Location, Activity of People, Places, Things
- Who? Where? When? What? Why?

User notes

slide #2

Slide text

Identity, Location, Activity of People, Places, Things

User Memo

context widgets

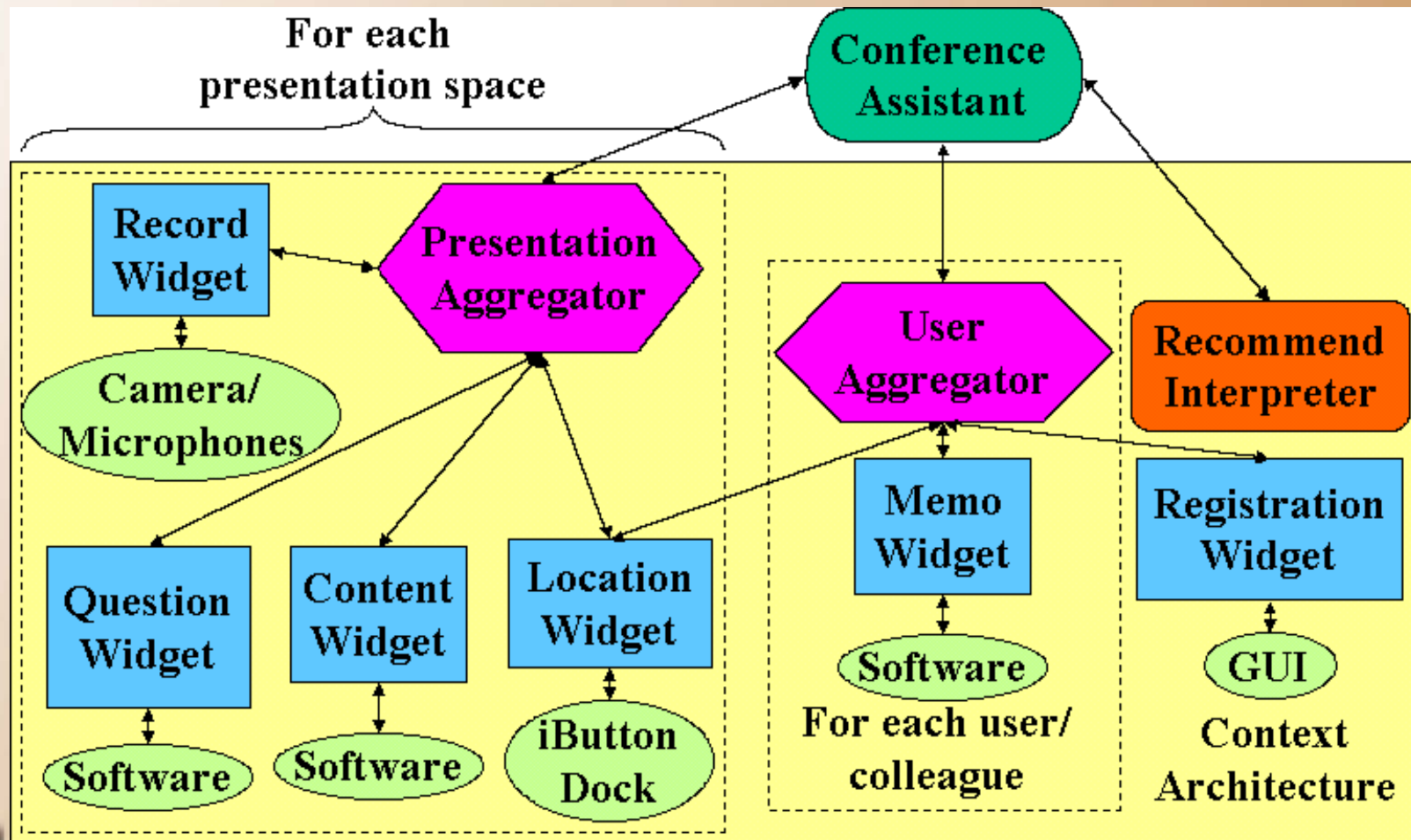
Back Forward

9:00  
Context  
VR Work  
VR Go

Devices  
Tracking  
art Floor

# Conference Assistant Arch.

- Complex application: **reuse, evolution**



# Outline

- Motivation
- Problems dealing with context
- Contribution: Context Toolkit
- **Validation:**
  - Design space and applications
  - **Building more realistic applications**
- Conclusions and future work

# Complex and Realistic Applications

- Privacy: Dynamic Door Displays
- Ambiguity: In/Out Board extension
- Security: Service/Context Access  
(SACMAT 2001)
- End-user programming: CybreMinder  
(HUC 2000)

# Component Abstraction

Features	Abstractions	
	None	Component
Context acquisition	X	P
Distributed communications	X	V
Query/subscribe	X	V
Storage	X	V
Multiple Apps	X	V
Sensor addition	X	P
Sensor failure	X	P
Evolution	X	P
Context specification	X	P
Situation realization	X	X

**Level of Support**  
**X** - none  
**P** - partial  
**V** - complete

# Component Abstraction

Features	Abstractions	
	None	Component
Context acquisition	X	P
Distributed communications	X	V
Query/subscribe	X	V
Storage	X	V
Multiple Apps	X	V
Sensor addition	X	P
Sensor failure	X	P
Evolution	X	P
Context specification	X	P
Situation realization	X	X

**Level of Support**  
**X - none**  
**P - partial**  
**V - complete**

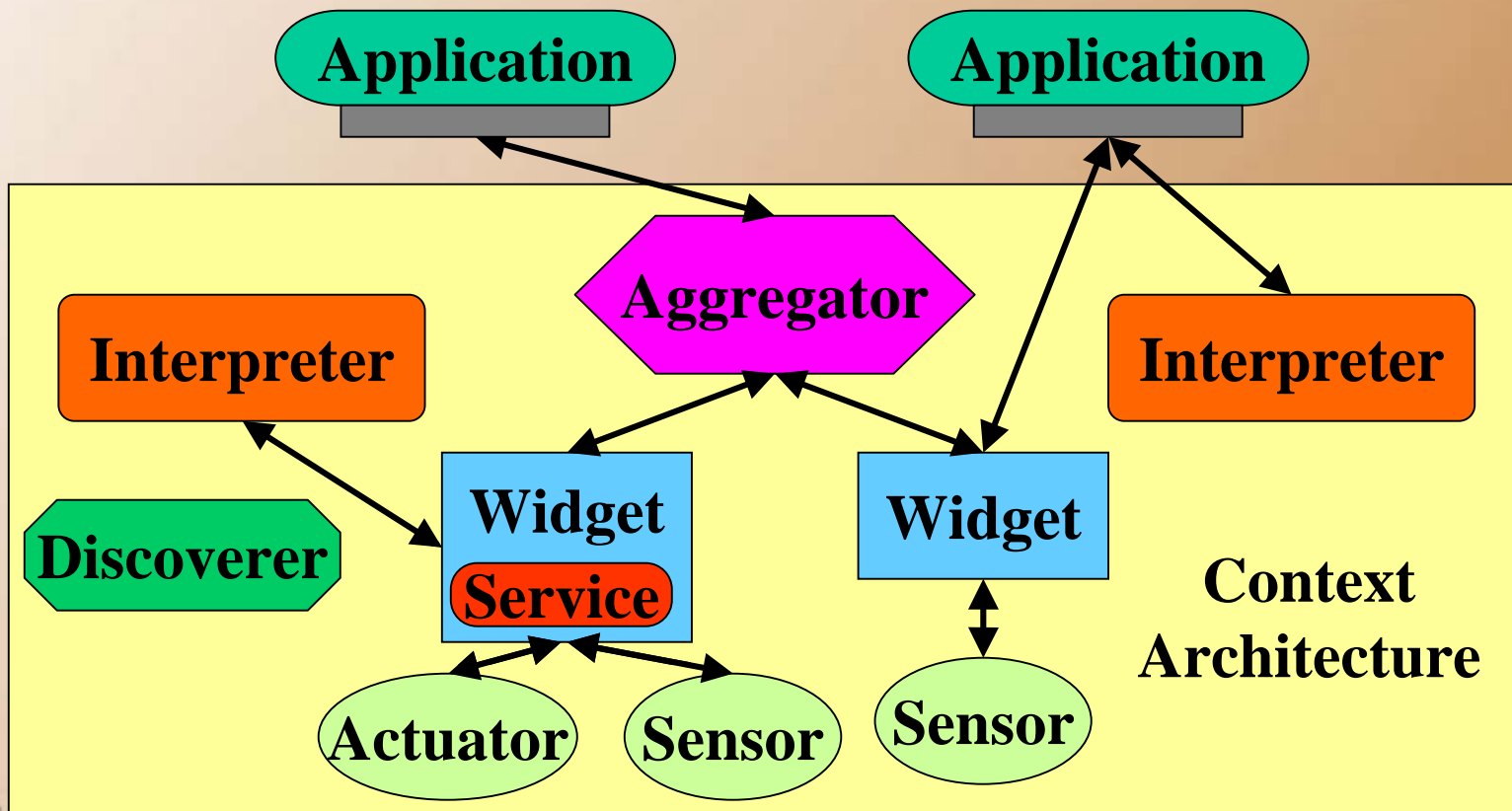
# Situation Abstraction: Declarative Style

- Revisit context definition
  - Allow programmer to define a situation (real-world callbacks)
- *Declare* what context you want, not how to obtain it
- Architecture's responsibility to deliver it
- Makes *specification* in design process simpler, more *robust*, easier to *evolve*



# Revised Framework

- Supports blackboard/box model of the world
- Different than component model



9/17/2001

# Situation Abstraction

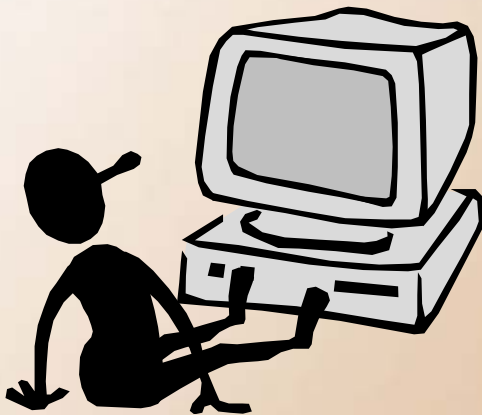
Features	Abstractions	
	None	Component
Context acquisition	X	P
Distributed communications	X	V
Query/subscribe	X	V
Storage	X	V
Multiple Apps	X	V
Sensor addition	X	P
Sensor failure	X	P
Evolution	X	P
Context specification	X	P
Situation realization	X	X

# Situation Abstraction

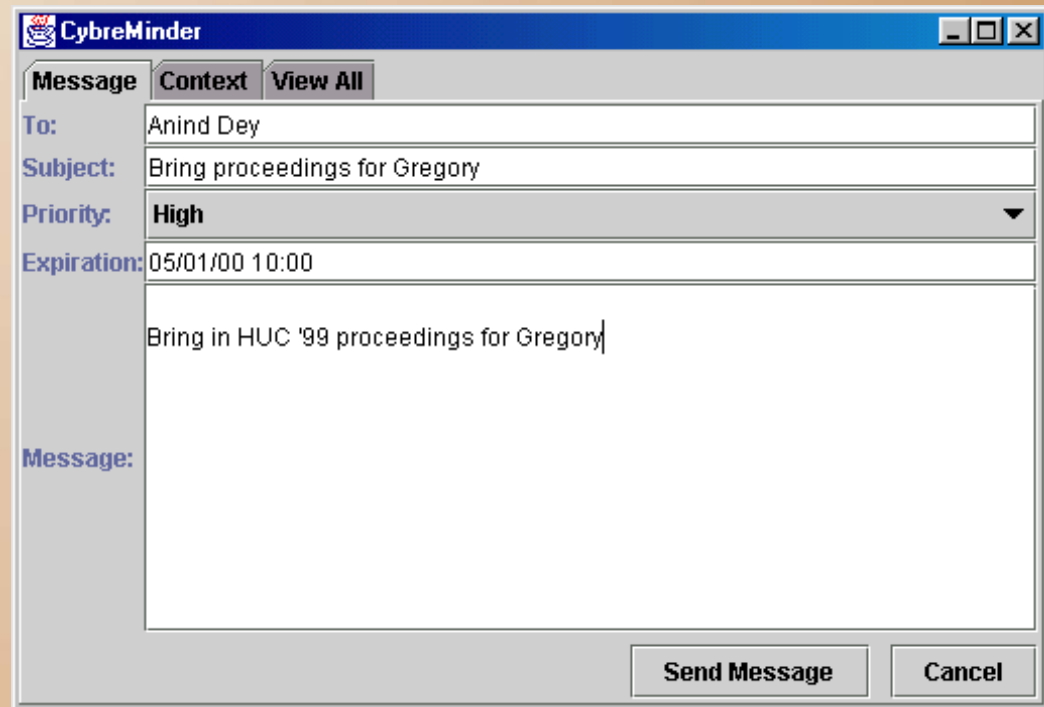
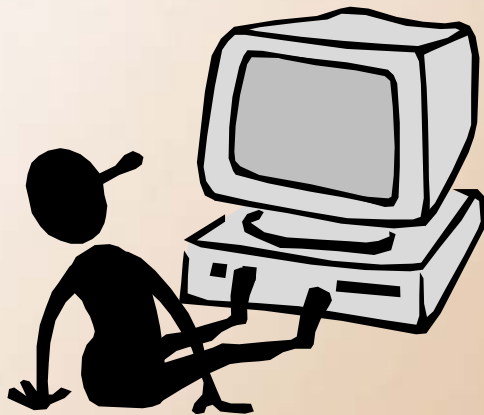
Features	Abstractions		Situation
	None	Component	
Context acquisition	X	P	P
Distributed communications	X	V	V
Query/subscribe	X	V	V
Storage	X	V	V
Multiple Apps	X	V	V
Sensor addition	X	P	V
Sensor failure	X	P	V
Evolution	X	P	V
Context specification	X	P	V
Situation realization	X	X	V

# CybreMinder (HUC '00)

# CybreMinder (HUC '00)



# CybreMinder (HUC '00)



# CybreMinder (HUC '00)

# CybreMinder (HUC '00)





# CybreMinder (HUC '00)



# CybreMinder (HUC '00)



**CybreMinder** [Window Title Bar]

Message Context View All

Available Sub-Situations

- A: timestamp, username, location | location=CRB
- city=Atlanta, weatherforecast=?, timestamp=?
- city=Atlanta, temperature=?, timestamp=?
- stockname=?, stockvalue=?, timestamp=?
- username=?, location=CRB, timestamp=?**
- location=CRB Conference Room, soundlevel=?, timestamp=?
- username=?, activitylevel=?, timestamp=?
- officename=CRB383, doorstatus=?, timestamp=?

Edit

Current Situation For Message

officename=CRB383, doorstatus=open  
username=Gregory, activitylevel=low

Clear Situation



# CybreMinder (HUC '00)

# CybreMinder (HUC '00)

**Reminder: Bring proceedings for Gregory**

**To:** Anind Dey

**Subject:** Bring proceedings for Gregory

**From:** Anind Dey

**Priority:** Normal

**Message:** Bring in HUC '99 proceedings for Gregory.

**Situation:** officename=CRB383, doorstatus=open  
username=Gregory, activitylevel=low  
username=Anind Dey, location=CRB

**Status:** delivered

**Submit Status** **Cancel**

# CybreMinder (HUC '00)



**Reminder: Bring proceedings for Gregory**

**To:** Anind Dey  
**Subject:** Bring proceedings for Gregory  
**From:** Anind Dey  
**Priority:** Normal

**Message:** Bring in HUC '99 proceedings for Gregory.

**Situation:**  
officename=CRB383, doorstatus=open  
username=Gregory, activitylevel=low  
username=Anind Dey, location=CRB

**Status:** delivered

# CybreMinder (HUC '00)



**Reminder: Bring proceedings for Gregory**

To: Anind Dey  
Subject: Bring proceedings for Gregory  
From: Anind Dey  
Priority: Normal

Message: Bring in HUC '99 proceedings for Gregory.  
officename=CRB383, doorstatus=open  
username=Gregory, activitylevel=low  
username=Anind Dey, location=CRB

Situation:

Status: delivered

Submit Status Cancel



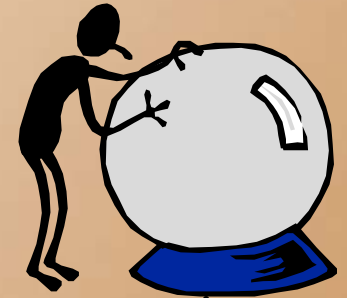
# Outline

- Motivation
- Problems dealing with context
- Contribution: Context Toolkit
- Validation:
  - Design space and applications
  - Building more realistic applications
- Conclusions and future work

# Research Contributions

- Conceptual framework requirements
  - Provide framework for designing apps more easily
  - Lower threshold to enable more designers
- Context Toolkit
  - Implementation and exploration of design space
- Support investigation of complex problems and more realistic apps
  - Raise ceiling
  - Privacy, uncertainty, security, end-user programming





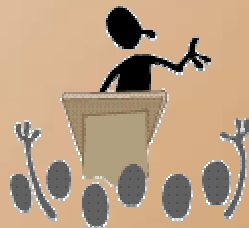
# What's Next?

- Complex interpretation, sensor fusion, dealing with ambiguity → how to infer what a user really wants
- Ontology, QoS, privacy, security
- Data models
- Development environment
- Evaluation of context-awareness
- What to do, and when and why
- Overload/how interruptible is the user
- End-user control of what happens
- Broaden scope of framework to be a general model of interactive and ubiquitous computing: look at implicit output

# Acknowledgements

- Gregory D. Abowd & FCE
- Motorola & NSF
- Contact info:
  - [anind@cc.gatech.edu](mailto:anind@cc.gatech.edu)
  - <http://www.cc.gatech.edu/~anind>
  - <http://www.cc.gatech.edu/fce/ctk>

- Questions?



# Intercom System

- Facilitate communications between family members:
  - In the same house
  - Between houses
  - While mobile
- Uses 4 types of context in combination (primarily present, service, but potential for tagging when learning)
- Leverage social mediation skills

# Design Process

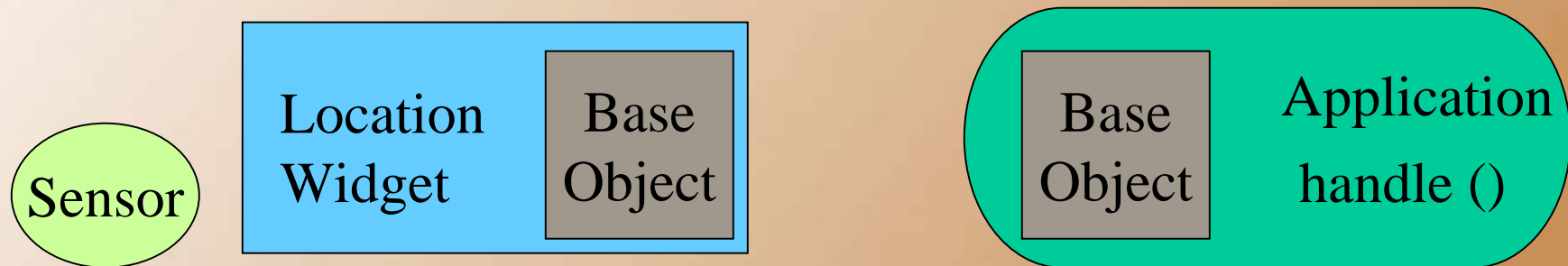
1. **Specification** - context and behaviors
2. **Acquisition** - install, API, query/notify, store, interpret
3. **Delivery** - deliver context to multiple, remote applications
4. **Reception** - locate relevant sensors, request context, interpret
5. **Action** - analysis and action

# How to simplify?

- Brooks 87: "No Silver Bullet: Essence and Accidents of Software Engineering"
  - essential problems
    - inherent problems
    - specific to the task at hand
  - accidental problems
    - problems induced by design tools
    - not specific to the task

# Callback Model

- Transparent communications, always available
- Similar to GUI architectures



# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

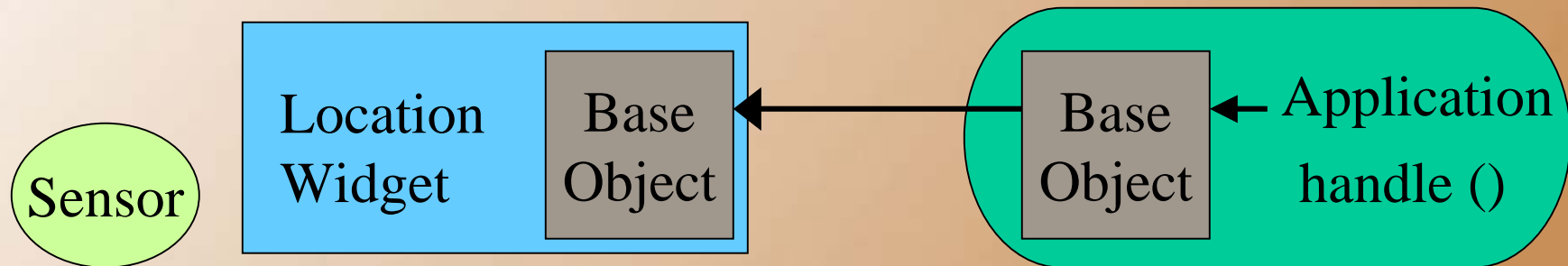
(a) subscribe: David  
in Room 343



# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

(a) subscribe: David  
in Room 343

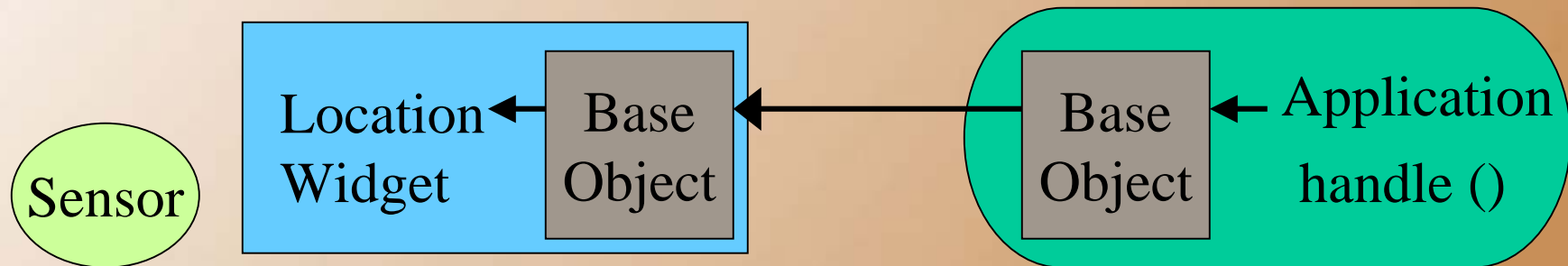




# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

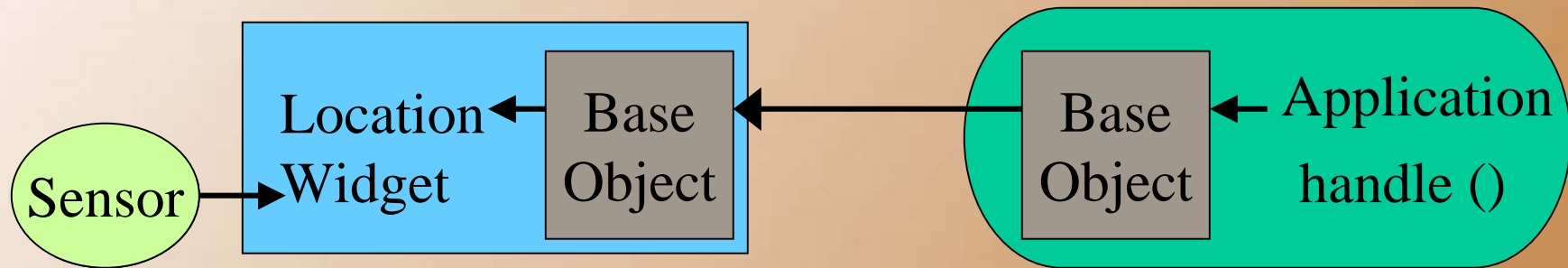
(a) subscribe: David  
in Room 343



# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

(a) subscribe: David  
in Room 343

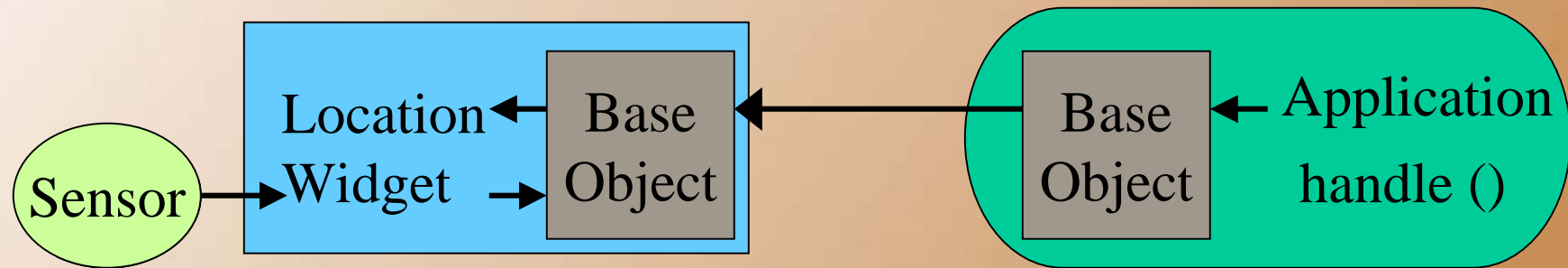


(b) sensor  
data arrives

# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

(a) subscribe: David  
in Room 343

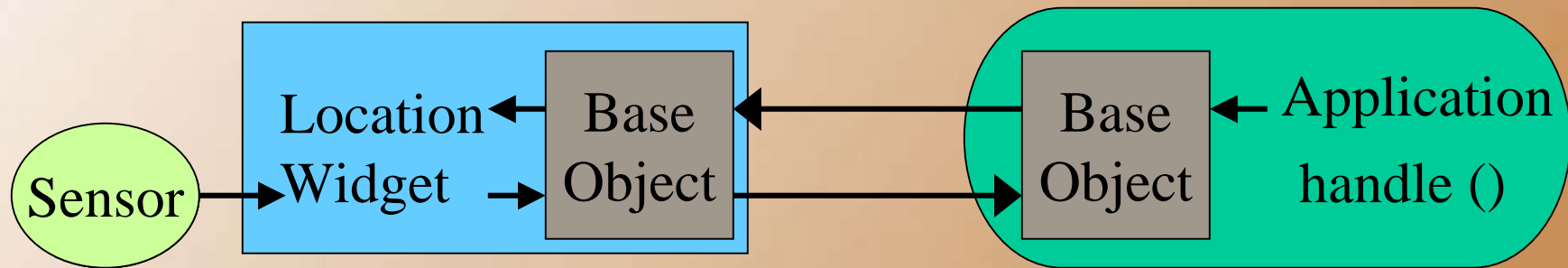


(b) sensor  
data arrives

# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

(a) subscribe: David  
in Room 343



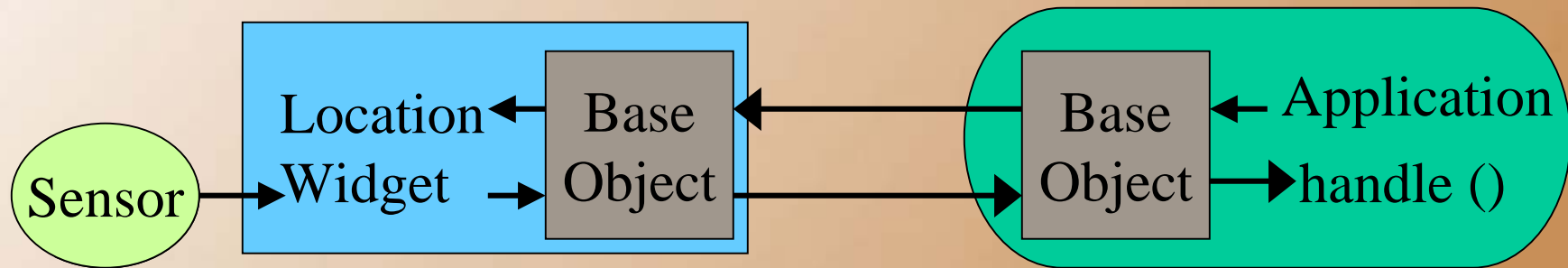
(b) sensor  
data arrives

(c) callback if  
data matches:  
David in 343

# Callback Model

- Transparent communications, always available
- Similar to GUI architectures

(a) subscribe: David  
in Room 343



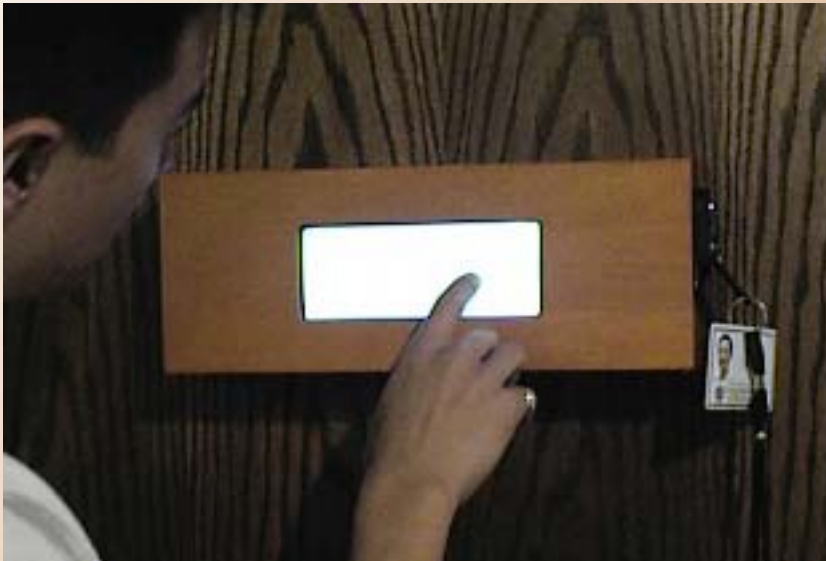
(b) sensor  
data arrives

(c) callback if  
data matches:  
David in 343

(d) callback  
delivered to  
a handler

# Access Control

- Dynamic Door Displays



# Access Control

- Dyn



**Everyday Computing Lab**

**Who are you here to see?**

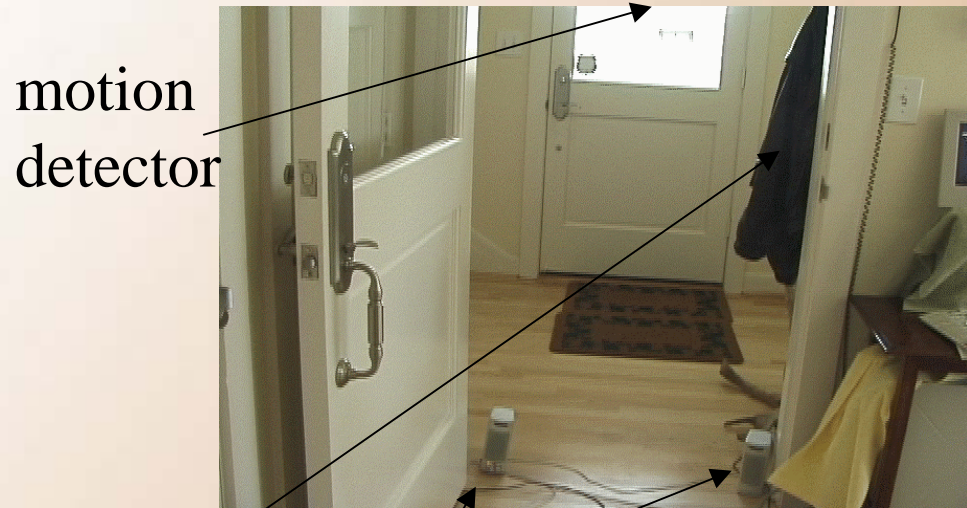
**Beth is in her office (CoC 256)**

**Leave Message**

**View Calendar**

**Back**

# Ambiguous Context



Who's home?

Goodbye Gregory Abowd

Please dock, speak, or type if this is wrong

Cory Kidd	Out 5:59pm
Jen Mankoff	Out 5:59pm
Rob Orr	Out 5:59pm
Brad Singletary	Out 5:59pm
Khai Truong	Out 5:59pm
Anind Dey	Out 5:59pm
Tanisha Hall	Out 5:59pm
Kent Lyons	Out 5:59pm
David Nguyen	Out 5:59pm
Daniel Salber	Out 5:59pm
Randy and Steve	Out 5:59pm
Gregory Abowd	Out 5:59pm

Enter your name



# Experiences: Limitations

- Continuously changing context
- Dealing with unreliable context and other quality of service issues
- Component failure
- Privacy



# Important Distinction

# Important Distinction

- Behavior that “looks easy” but is not.
  - Star Trek's doors
  - Real-time classroom control
  - Incremental speech recognition improvement

# Important Distinction

- Behavior that “looks easy” but is not.
  - Star Trek's doors
  - Real-time classroom control
  - Incremental speech recognition improvement
- Behavior that “looks hard” but is not.
  - Mobile tour guide (GPS, IR beacons)
  - Temporal synching

# Distribution of Sensing

- Heterogeneity of platforms and languages
  - No guarantees on what sensors require
  - No guarantees on what what's available
  - No guarantees on what developers prefer

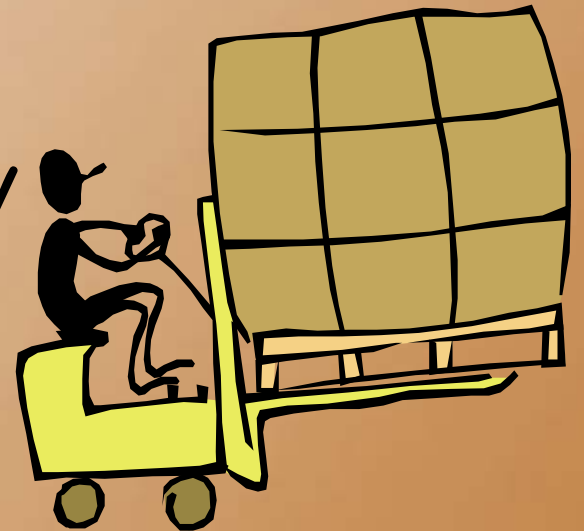
# Abstraction: Interpretation

- Provide meaning to sensed data
- Simple converters
- Complex inferences



# Abstraction: Aggregation

- Eases interpretation
- Maps to notion of an entity
- Efficiency mechanism



# Component Persistence

- Not like GUI widgets
- Execute autonomously
- Always running





# Context History

- Not like GUI widgets
  - Don't want to leave to apps
- Components always running
  - Store data for future apps



# Situations: Declarative Style

- Say what you want, not how you want it done - framework figures it out
- Allow programmer to define a situation (complex real-world callbacks)
- *Specification* in design process simpler
- More robust w.r.t. component failures and easier to evolve