MOBILE COMPUTING
CSE 40814/60814
Fall 2015

System Structure

explicit input \[\rightarrow\] System \[\rightarrow\] explicit output
Context as **Implicit Input**

![Diagram](image)

**Context:**
- state of the user
- state of the physical environment
- state of the computing system
- history of user-computer interaction
- ...

---

**What is Context?**
Examples of Context

- Identity (user, others, objects)
- Location
- Date/Time
- Environment
- Emotional state
- Focus of attention
- Orientation
- User preferences
- Calendar (events)
- Browsing history
- Behavioral patterns
- Relationships (phonebook, call history)
- ... the elements of the user’s environment that the computer knows about...

Relevance of Context Information

- Trying to arrange lunch meeting
- Going to a job interview
- Going home after work and making evening plans
- Shopping
- Tourist
- ...
Examples

• Smartphone adjusts the screen to the orientation of the device
• Apple Watch turns on display if arm lifted/rotated
• Orientation is determined by using both a gyroscope and an accelerometer.

Examples

• Phone display adjusts the brightness of the display based on the surrounding area
• Uses a light sensor
Examples

• Device uses GPS to display the user’s location
  • Can use to find nearby stores
  • Get directions
  • Location-specific status updates on social media

Examples

• The time is displayed on the phone.
  • Time zone change
  • Daylight savings time
Examples
• Device disables touch screen when the user speaks on the phone
• Uses a proximity sensor (infrared signal travel time)

Examples
• Active Badge location system
  • One of the first context-aware applications
  • Context = location
• Call-forwarding system
• Issues
  • Private call forwarding to a public room
  • Call is forwarded to important meeting
Examples

- Schneider trucking trackers
  - Uses GPS to track loads
  - Sends a notification when a load nears its destination
  - Sends emergency notifications when conditions are met

Definitions of Context

- Example: (Train) booking application
  - Customer number, booking details are required and must be provided by the user (explicit input)
  - Location, time are required and can be automatically derived from context information (implicit input)
  - There is additional context information (temperature, …) that is not relevant for the application
Definitions of Context

- By enumeration

“Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves” [Dey et al. 2001]

- Most prominent definition
Definitions of Context

• Functionality & Relevance:
  • Context characterizes the actual situation in which the application is used. This situation is determined by information which distinguishes the actual usage from others, in particular characteristics of the user (her location, task at hand, etc.) and interfering physical or virtual objects (noise level, nearby resources etc.).
  • Thereby, we only refer to information as context that can actually be processed by an application (relevant information), but that is not mandatory for its normal functionality (auxiliary information).

Classification

• External (physical)
  • Context that can be measured by hardware sensors
  • Examples: location, light, sound, movement, touch, temperature, air pressure, etc.

• Internal (logical)
  • Mostly specified by the user or captured monitoring the user’s interaction
  • Examples: the user’s goal, tasks, work context, business processes, the user’s emotional state, etc.
Why Use Context?

- **Reduce cognitive load of user**

- **Proactivity**
  - Set up environment according to user’s preferences or usage history
  - Auto-completion of forms (location, time in timetable)
  - Reminders

- **Search and filter information** according to the user’s current needs

- **Avoid interrupting** the user in inappropriate situations

- **Smart environments**
  - Turn devices on/off, start applications, ... depending on location, time, situation (lecture, meeting, home cinema, ...)
  - Discover and use nearby interaction devices

Why Context-Aware Computing?

<table>
<thead>
<tr>
<th>Existing Examples</th>
<th>Context Types</th>
<th>Human Concern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Lights On / Off</td>
<td>Room Activity</td>
<td>Convenience</td>
</tr>
<tr>
<td>File Systems</td>
<td>Personal Identity &amp; Time</td>
<td>Finding Info</td>
</tr>
<tr>
<td>Calendar Reminders</td>
<td>Time</td>
<td>Memory</td>
</tr>
<tr>
<td>Smoke Alarm</td>
<td>Room Activity</td>
<td>Safety</td>
</tr>
<tr>
<td>Barcode Scanners</td>
<td>Object Identity</td>
<td>Efficiency</td>
</tr>
</tbody>
</table>
Why Context-Aware Computing?

<table>
<thead>
<tr>
<th>Other Examples</th>
<th>Context Types</th>
<th>Human Concern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Cell Phone Off In Meetings</td>
<td>Identity</td>
<td>Convenience</td>
</tr>
<tr>
<td>Tag Photos</td>
<td>Time Location</td>
<td>Finding Info</td>
</tr>
<tr>
<td>Proximal Reminders</td>
<td>Proximity</td>
<td>Memory</td>
</tr>
<tr>
<td>Health Alert</td>
<td>Activity</td>
<td>Safety</td>
</tr>
<tr>
<td>Service Fleet Dispatching</td>
<td>History</td>
<td>Efficiency</td>
</tr>
</tbody>
</table>

Categories of CA Applications

<table>
<thead>
<tr>
<th></th>
<th>Manual</th>
<th>Automatic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Getting Information</td>
<td>Proximate Selection &amp; Contextual Information</td>
<td>Automatic Contextual Reconfiguration</td>
</tr>
<tr>
<td>Executing Command</td>
<td>Contextual Commands</td>
<td>Context-Triggered Actions</td>
</tr>
</tbody>
</table>
Proximate Selection/Contextual Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Room</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>caps</td>
<td>35-2200</td>
<td>200ft</td>
</tr>
<tr>
<td>claudia</td>
<td>35-2108</td>
<td>30ft</td>
</tr>
<tr>
<td>perceptor</td>
<td>35-2301</td>
<td>20ft</td>
</tr>
<tr>
<td>snoball</td>
<td>35-2103</td>
<td>100ft</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Distance</th>
<th>Name</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>20ft</td>
<td>perceptor</td>
<td>35-2301</td>
</tr>
<tr>
<td>50ft</td>
<td>claudia</td>
<td>35-2108</td>
</tr>
<tr>
<td>100ft</td>
<td>snoball</td>
<td>35-2103</td>
</tr>
<tr>
<td>200ft</td>
<td>caps</td>
<td>35-2200</td>
</tr>
</tbody>
</table>

Table 2: UI Techniques for Proximate Selection

Bluetooth

- Discovery Results
  - Nearby devices
  - Select a PC:
    - L000S92Y6F1
    - PALMONE-0W96V71
    - Jim's PC
    - PALMONE-71ZAG
    - StenDellBlue
    - Bonnie

- Devices
  - M2000 by Plantronics
  - DELL AXIM X50
  - iPAQ 2215
  - MacMini

- Options:
  - 1 New
  - 2 Edit
  - 3 Delete
  - 4 Set as Hands-free

Done Menu
Automatic Contextual Reconfiguration

- Add, remove, or alter components based on context
- Smart notifications on phone (ring, vibrate, autoresponse)

Contextual Commands

- Users can parameterize commands with context-filtered values; execution changes based on context
- Example: universal remote control
Context-Triggered Actions

• Simple if-then condition-action rules, automatically invoked
• Reminder: if I step into the car on weekday morning and don’t have suitcase with me, remind me to get it

• CybreMinder:

Context-Triggered Actions

• Challenges:
  • Expressiveness of language for rules
  • Accuracy of context information
• Siren:
Context-Awareness: Risks

- **Context-awareness helps technology to “get it right”**
- But context is hard to sense (quantity, subtleness)
- Computers are not self-aware like humans

- **Problems:**
  - When the system does the wrong thing
    - auto-locking car doors
    - screen saver during presentation
    - microphone amplifying a whisper

- **Context data must be coupled with the ability to interpret it, but computers are bad at “common sense”**.
- More rules ≠ intelligence
- More rules = more complexity, harder to understand

- **“Human in the Loop”:**
  - computers can detect, aggregate, portray information
  - allow human users to interpret and act on it
  - is this a good strategy for all context-aware systems?
5 Design Considerations

1. Improving relevance
   - Deciding when a communication is relevant to the person’s current (or near future) situation.
   - For example, getting notification about an email from your travel agent regarding itinerary changes while packing to leave for the airport.

2. Minimizing disruption
3. Improving awareness
4. Reducing overload
5. Selecting channels

---

5 Design Considerations

1. Improving relevance

2. Minimizing disruption
   - Deciding when and how to notify people that they have a communication.
   - For example, your phone should vibrate and not ring, when you are at the symphony (unless it is truly urgent).

3. Improving awareness
4. Reducing overload
5. Selecting channels
5 Design Considerations

1. Improving relevance
2. Minimizing disruption
3. Improving awareness
   • Deciding what information and mechanisms can help people make intelligent communication decisions.
   • For example, the caller should be told you are at the movies before the call goes through.
4. Reducing overload
5. Selecting channels
5 Design Considerations

1. Improving relevance
2. Minimizing disruption
3. Improving awareness
4. Reducing overload
5. Selecting channels
   • Deciding which communication device should be used to get in touch with somebody.
   • For example, routing calls to your home phone instead of your cell phone when you are at home and cellular reception is poor.

Building Context-Aware Systems/Apps

• Specification: What context-aware behavior should be implemented? Which context is required for that purpose?
• Acquisition: Which sensors can be used to retrieve this context?
  → Context Sources
• Delivery and Reception: How is the context represented, managed and exchanged?
  → Context Models
  → Access Mechanisms
  → Context Storage and Management
• Action: Which actions should be taken corresponding to the captured context?
Context Sources

- **Sensed context**
  - query physical sensors or applications (virtual sensors)
  - Examples: temperature, calendar entries
- **Inferred or derived context**
  - combining context data to gain new information ("higher level context")
  - examples: mobility (e.g., "leaving office") + time (e.g., "12.00pm") = "lunch break" (activity)

<table>
<thead>
<tr>
<th>Context Type</th>
<th>Sensors</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensed context</td>
<td>Physical sensors</td>
<td>Temperature</td>
</tr>
<tr>
<td></td>
<td>Virtual sensors</td>
<td>Calendar</td>
</tr>
<tr>
<td>Inferred Context</td>
<td>Logical Sensors</td>
<td>Activity</td>
</tr>
</tbody>
</table>

Context Models

- Context data must be represented in machine readable form to enable application to use it
- Context model defines exchange of context information
- Context model has to provide a useful set of attributes for each context data (type, value, timestamp, source...), ideally it addresses how to cope with incompleteness and ambiguity of context information

- Existing Context Models can be classified by means of the data structure they use for exchanging context information:
  - Key-Value Model
  - Markup Scheme Model
  - Object-oriented Model
  - Logic-based Model
  - Ontology-based Model
Accessing Context

• Two ways of getting informed of context data:
  • Queries: request context information
  • Event Subscription: the actual applications are notified every time a specified event occurs

• Consider Privacy and Security concerns, for example by
  • Specifying domain dependent policy rules for access control
  • Allowing the user to control the access to his context data

Context Storage and Management

Context storage and management
• Specify a well-defined interface for accessing the context data
• Answer queries and notify the actual applications of context changes
• Maintain a context history or at least a context buffer
• Provide a discovery services for the various context sources
Challenges & Issues

- Developers have **little experience with devices that gather the data** (e.g., gyroscopes).
- Data gathered from a sensor **must be interpreted correctly** in order for it to be useful.
- Context comes from various sources and in order for this data to be useful it **must be combined correctly** (i.e., the gyroscope and accelerometer working together to determine orientation).
- The context **changes constantly in real time**.

Challenges & Issues

- **Increasing usability or decreasing control?**
  - **Automation** reduces the amount of work that users have to do
  - Users like the idea of a device that completes tasks on their behalf
  - However, when users use these devices they feel a **loss of control** if a device has a high level of automation
Challenges & Issues

• **Privacy issues**
  • Should law enforcement be able to access the history of a user?
• Context recognition is **not always correct**
  • Errors fusing data
  • Detection errors
  • Interpretation errors
• Application becomes too **complex**
  • Reduces accuracy of the application

Challenges & Issues

• **User preferences may not match what the device does!**
  • Everyone is different!
    • What is your idea of “nighttime”?
    • What is your idea of “warm”? Or “loud”?
• **Information overload**
  • Can overwhelm the user
Solutions

• Keep an appropriate level of automation (avoid uncertainty)
  • The more automation we have, the less control we have over what is happening.
  • What happens if we give all control to machines?
  • Would you trust your phone to give you a dose of medicine?
  • Keep a balance between uncertainty and automation.

Solutions

• Avoid unnecessary interruptions
  • Phone flashes a notification every 30 seconds
  • Eventually the user will ignore it!
• Avoid information overload
  • Too much information can overwhelm the user, and bog down the device
  • Example: Walking down a busy street a user’s device is bombarded with suggestions of places to shop
Solutions

- Be sure the user feels in control of the application
  - Too many automated actions = loss of control
  - A balance between automation and user control needs to be found
- Example: Smart Actions App
Solutions

• Keep an appropriate level of system status visibility
  • Allow the user to see what action the device is taking
  • Be sure the user understands why the device is performing the action
• Account for the impact of Social Context
  • A loud alert is not ideal for all situations
• Allow for the personalization of individual needs
  • Allow user to change location names (set a location name to “home” for example)

Solutions

• Secure the user’s privacy
  • Selling information to advertisers…is this right?
  • Giving information to the police, when does this cross the line?
  • Sharing context information with others—Facebook location
Future

- Healthcare
  - Detecting problems, notifications, interventions
- Retail & Restaurants
  - Recommendations, alerts, diet considerations
- Vehicular
  - Accident prevention, driver fatigue, car customization
- Gaming
  - Motion sensing, virtual reality
- Entertainment
  - Recommendations, history, alerts

- [INSERT YOUR IDEA HERE]