Context-aware Computing (Chapter 4)
As human beings, we are adept in our understanding and use of context in our daily activities. We routinely use contextual information, such as who is in our vicinity or where we are, to modulate our responses to or interactions with other people. As we mature, we learn which contextual information is important given a particular situation. For example, when we are with our close friends, we rarely worry about how we speak: “Like, you know what I mean.” However, in a formal situation, we are careful in what we say and how we say it.
Context-aware Computing

- Beyond application-aware adaptation
- Instead of adapting only to resource levels, adapt to contexts

Context:
- Enumeration-based (categories)
- Role-based (roles of context in building mobile applications)
Types of Enumeration Based Context

- **Computing context** includes network connectivity, communication costs, communication bandwidth, and local resources, such as printers, displays, and workstations.
- **User context** includes user profiles, location, and people in the vicinity of the user.
- **Physical context** includes lighting and noise levels, traffic conditions, and temperature.
- **Temporal context** includes time of day, week, month, and season of the year.
- **Context history** is the recording of computing, user, and physical context over time.
The 5 W’s…

- **Who** is the user? Who are the people with which the user is interacting, or who is nearby?
- **What** is the user doing?
- **Where** is the user? Home? Work? Bathroom? Familiar coffee shop?
- **When?** What time is it?
- **Why?** Why is the user performing a certain task? What is the task’s priority in the “grand scheme”?
- **Low-level vs. High-level details**
Context Overview
Role Based Context

Finding role of context in designing applications

- Active Context: contextual information used by the application to adapt its behaviour.
- Passive Context: contextual information that is not critical for application adaptation but is provided to the user to enhance his or her understanding of the situation.
Context-Aware Computing & Applications

- Context-Aware Computing Devices & Applications - Mobile applications
  - User’s context changes frequently
  - Need context-aware behaviour

- Capability & Features
  - Proactive in acquiring contextual information
  - Adapt their response based on the acquired info
Context-Aware Computing & Apps

**Tasks:**

- Providing a contextual user interface
- Presenting contextual info
- Presenting non-contextual info
- Context sensitive info services
- Proactive context-aware adaptation of behavior (auto reconfiguration). It is automatically initiated by system or Appl.
- Reactive context-aware adaptation of behavior (enhancing situation understanding). It is in response to user request
Context-aware Capabilities and Requirements

- **Contextual sensing**
  - detection of environmental states and presenting it in user-friendly form. For example, the location information obtained from a location sensor like GPS satellite can be presented to the user via a map with a “you are here” marker instead of latitude, longitude and altitude.

- **Contextual adaptation**
  - capability of the system to adapt its behavior by using contextual information

- **Contextual resource discovery**
  - capability to discover available resources in an environment and use it to better adapt to the user’s need.
✓ **Contextual augmentation**
  - capability to associate contextual information with some digital data
  - Example: association of a particular meeting place and attendees with a set of minutes
  - Example: association of a digital photo with a specific location
Types of Context Awareness App.

- Function or Service Type
  - Providing information:
    - Contextual: selection, presentation & info.
  - Actuating commands: commands, actions & tagging
- Initiating Agent
  - Manual: initiated explicitly by
  - Auto: Invoked implicitly by application
- Adaptation (contextual selection)
  - Information
  - System
  - User Interface
Developing Context-Aware Applications

1. Identifying relevant context – application dependent
2. Specifying context-ware behaviours: considering reuse
3. Integrating with mechanisms for acquisition of contextual information – platform dependent
Designing CA Applications

✓ Build list of relevant contexts
  – e.g., “home”, “office”, “traveling”, “sleeping”, …
✓ Specify context-aware behaviors
  – Presentation of context-sensitive information
  – Automatic discovery of relevant objects (e.g., nearby people for transmission of business cards)
  – Modification of the physical and digital environments
✓ Integration of application with methods for sensing context
Specifying Context-aware Behaviors

- Reuse
- Two Example Approaches
  - Context-triggered actions
  - Stick-E notes
Specifying Context-aware Behaviors: example 1

- **Context-triggered actions**: Watchdog and contextual reminder for active badges
- Watchdog program is designed for UNIX environment coupled with an Active Badge Location System.
- Active Badge Location System: gives information about the mobility of the wearer.
  - Watchdog monitors Active Badge activity
  - and executers relevant shell commands as required.
  - System configuration files – parameters, (location, action): contains description of Active Badge events and actions to perform.
  - Watchdog Monitoring Activities & Events:
    - Arriving, Departing, Settle-In, Missing
Specifying Context-aware Behaviors: example 2

- Stick-E Note
  - motivated by Post-It note

- Supporting platform
  - For PDA with wireless connectivity to a communication network
  - Equipped with various sensor: GPS, etc

- Major components
  - Context
    - Location, nearby users, time (where, who, when)
  - Content
    - Information, Actions, Interfaces
**Simple Example: stick-e notes**

- Context-aware Post-it notes
- Build list of relevant contexts
  - Based on location (latitude/longitude via GPS)
  - Temperature, whatever else can be sensed
- Specify context-aware behaviors
  - stick-e notes “pop up” on a PDA when contextual info is appropriate
  - Reminder to return a library book when near the library
  - Reminder to buy a new winter jacket when temperature drops below 60F
- Integration of application with methods for sensing context
  - Ubiquitous sensing of environmental characteristics, such as location, temperature, number of human beings nearby, the cat is near, not widespread
Where Does Context Come From?

- Environmental sensors
  - Temperature, humidity, location, noise, motion
  - CAT sensor (context aware toolkit)
  - Potential need for multiple types of sensors
  - GPS vs. indoor location sensors

- History
  - Recording user actions and previous contexts

- User’s personal computing environment
  - Schedules, notes, address books, financial info

- Need real-time analysis to provide context
Example: Location

- Indoor locating systems
  - e.g., infrared or ultrasound
- Wireless nanocell communication activity
  - Association with short-range base stations
- GPS
- Associations with nearby computers
- Motion sensors and cameras, computer vision
- Ask the user!
Middleware Support

Challenges - Context-aware applications

1. Contextual info – from various heterogeneous and distributed sources
2. Same type of contextual info may have to be obtained from different sources at different times
3. The low-level contextual info must be abstracted to be useful
4. Context-awareness is most relevant when the environment is highly dynamic (mobile)
Middleware Support

Contextual info – from various heterogeneous and distributed sources

- Hardware and software sensors: motion detectors, noise, temperature sensors, location systems
- System recorded input – user-system interaction history
- Other applications
  - User’s personal computing space:
    - Schedules, calendars, address books, contact lists, and to-do lists
  - Distributed computing environment
    - Obtained from applications running in the vicinity of these devices: shopping malls, freeway, etc
Middleware Support

2. Same type of contextual info may have to be obtained from different sources at different times
   - GPS receiver – outdoor positioning system
   - Indoor positioning system

3. The low-level contextual info must be abstracted to be useful
   - GPS position info (latitudes and longitudes)
   - → Tour guide, location info

Context-awareness is most relevant when the environment is highly dynamic (mobile)
   - Real-time detection, trigger commands, auto reconfiguration
Middleware Support – Contextual Services

- Middleware Infrastructure that can provide the following services (acquire contextual input from sensors)
  1. Contextual subscription and delivery service
  2. Context query service
  3. Context transformation service
  4. Context synthesis service
  5. Discovery and management service

- Actuator Services
  - Perform a context-dependent output function
Middleware Support – An Example


Software components for context acquisition

- Context widgets
  - An interface between sensors and applications
  - Provides an abstraction layer
  - State: a set of attributes
  - Behavior: call back function
  - Persistent entities that can be shared by multiple applications
- Context interpreters (info translators)
- Context aggregators
  - Entity: person, room, software systems, hardware devices
- Discoverer
  - Discovery and management services
Key Points

✓ Context-Sensitive (Aware) computing is antithetical to traditional computer science (search for context-independent abstractions)
  – Black box approach
✓ Model of Context Aware application: Context as implicit input
  – System boundary decides what’s implicit and what’s explicit
✓ Context-Abstraction trade-off
Key Points (Cont.)

✓ Models of context:
  – System model: description of the system
  – User model: User’s state, history and preferences
  – Task model: goals and actions intended to be performed by the user

✓ To create context-aware applications these models should be dynamic and have ability to explain themselves.
Mobile Middleware (Chapter 5)
Introduction Mobile Middleware

- What is Mobile Middleware
- Adaptation [Chapter 1 of Sandeep K.S. Gupta]
- Agents [Chapter 6 of Sandeep K.S. Gupta]
- Service Discovery [Chapter 7 of Sandeep K.S. Gupta]
What is Mobile Middleware?

✓ Software that supports mediation between other software components, fostering interoperability between those components across heterogeneous platforms and varying resource levels.
What is Mobile Middleware?

- Mobile middleware allows for the implementation of distributed applications connecting mobile and enterprise applications over wireless networks.
- Provide the “black box” technology that connects mobile devices on the front lines of the enterprise to the back-end applications running on corporate servers.
3 major types of middleware for mobile computing:

1. Adaptation
   - it allows applications to offer reasonable performance to users across different environments.

2. Mobile Agent
   - it extend the reach of data servers & help mobile devices conserve energy.

3. Service Discovery
   - it allow mobile devices to change configuration quickly and easily, depending on available services.

Note: Context Aware Computing can also be encapsulated in middleware frameworks
Adaptation

Tasks

- Adapt behavior and expectations to conserve scare resources
- Adjust quality of service (QoS) – guarantee performance

How should adaptation be supported?
- Monitor resources and adapt appropriately

Eg. Video player
Agents

Allowing programs to move autonomously about a network in order to access remote resources

- Migrate to servers -> access data or computational resources -> migrate again -> return to home base

Benefits

1. Disconnection is easily supported
2. Access to large amount of data to solve problem
3. Allow the functionality of servers to be expanded dynamically
Service Discovery

- Extend the client-server paradigm
- Discover needed service on-demand
- Bluetooth Service Discovery Protocol