

CS 525M – Mobile and Ubiquitous Computing Seminar

Michael Theriault

A Context Aware User Interface for a Ubiquitous Computing environment at WPI

- After struggling to build a sensor network, I decided to switch to building a context aware user interface for a Ubiquitous computing environment at WPI.
- I am assuming that in a ubiquitous computing solution at WPI, students and faculty will have access to a wide variety of information.
 - Need to be able to sort the important information from the unimportant information.
 - Need to avoid overloading the user with information.
 - Need to provide the user with information when it is needed or useful.

Proposal

- Want to create a user interface for a ubiquitous computing environment at WPI.
 - Assuming a wide variety of state data is available from the network, such as time, location, weather.
 - Using the concept of an event for something that requires some user interaction of some sort. This could be anything from homework assignments to a beach volleyball game, to buying books and paying bills.

Context Aware Priority

- Priority of an action may change due to a change in context.
- The idea is to provide information when it is either useful or necessary.
- A simple example would be that you want to be informed when a party is happening right before the party starts, where the information is less useful if you get it a week before.

Previous Work

- This is very similar to the context aware hospital when a doctor would receive a medical chart when they enter a patients room.
- Also provide the doctors with new information as they travel the hospital.

Difference

- The difference is that in this environment, the information is always available to the user.
 - The context only makes the information more likely to be either received or ignored.
 - Also, a wide range on contexts could influence how much priority an event will get.
 - Example: The priority of buying textbooks could be increased by being in the bookstore. It could also be increase by classes starting.

Filtering

- Need to provide a flexible framework to allow users to filter out information that they don't want.
 - Important not to spam users with information that they do not care about.
 - Attempt to increase the signal to noise ratio of information

Filtering Continued

- Server needs to present information in a form that is easily filtered.
- Server information has the following fields
 - Person: Who is involved
 - Location: Where is it
 - Type: What category it belongs to
 - Date: When it is happening.
 - Priority: How important is it?

More Filtering

- Need to be able to filter based on any one of these criteria, or a combination.
- Example: I don't want to be informed of low priority academic events. Or, I don't want to be informed about low priority academic events that occur on a weekend.

Implementation

- The program is going to be written in Java.
 - Provides easy access to GUI widgets.
- Server occurs in different Frame than the user.
 - Removes the networking problem from the program and allows me to focus on the GUI
- The server can add events, the User can add filters. Both can see the data of the other.

More implementation

- The Server is responsible for maintaining and changing the context in this simulation.
- The User GUI then reacts to the change in context, and changes what is visible to the User.
- The system also has the ability to load both filters and events from a file at the beginning of the simulation.

Results

WPI Ubiquitous Computing System

Add Filter Change Context Sensitivity

User Alert Log

Location	Person	Type	Importance	Description	Date
school	Professo...	Academic	Important	Project P...	4/27/2004
home	Bob	Work	Medium	Meeting	4/28/2004
school	none	Academic	Medium	Sign up f...	5/30/2004

Filters

Priority: Low
Priority: Medium & Type: Social

Status

User: Mike
Location: school
Type: grad student

Concerns, conclusion

- The approach seems to be a reasonable approach to dealing with the large amount of data a system like this would generate.
- However, the GUI may need to be improved because it seems very text heavy, and that may be intimidating to the user
- Need to add context clues to the user, so that the data is easier to digest. Probably change the color of the text based on priority.

Future Work

- Could add multiple users to the system, and include the interaction between them in events.
 - For example, if two people have a meeting event, if they wind up at the same place at the same time, they had their meeting.
- Would like for the user GUI to assist in resolving certain events.
 - For example, if someone has a bill they have to pay, create a dialog box with payment options.

Questions?

Questions?

