Infrastructure for Multi-User, Multi-Spatial Collaborative Environments

Stephen Peters PORTNOY@PORTNOY.ORG
MIT Artificial Intelligence Laboratory, 200 Technology Square, Cambridge MA, 02139 USA

Recent research into intelligent environments (Nixon et al., 1999; Coen, 1998) has focused primarily on systems where a single user interacts with a single intelligent space, through the use of speech, gestures, or more mundane keyboard input. These systems often ignore handling multiple people, yet much interaction occurs in a conference or meeting room, with multiple people interacting with each other rather than with a computer presence. In addition, these research efforts usually limit themselves to a single intelligent environment. As the Oxygen project moves us towards a world where people move between intelligent spaces, we need to create spaces that can react differently based on the people who are within them, and which can work together to serve the needs of the users.

Attempts to achieve these capabilities without representing individuals as individuals per se, such as the KidsRoom project (Bobick et al., 1998), creates systems that can only recognize users as aggregate concepts, and are limited to very simple feedback mechanisms, without the ability to target the feedback to one user. The EasyLiving project(Brumitt et al., 2000) incorporates the notion of individuals into its system, and can track multiple independent people within its room. However, it is still tied closely to one specific space, and has yet to be extended to a multi-spatial world.

In this project, we take the existing Metaglue (Coen et al., 1999) intelligent environment system and extend its capabilities to handle multiple users and spaces with the addition of a simple knowledge representation, encompassing representations of users and spaces, and coupling it with an information base describing the capabilities of the spaces and roles for the users.

Approach

The approach taken here is to split the set of agents currently running in the system into groups (referred to as “societies” in the Metaglue system), based on the entity on whose behalf they are operating. For each of these agent societies, there is an “ambassador” agent, which contains information about the represented entity, and allows the entity to exert some control over the agent cloud. This division is not quite as clear-cut as it might seem at first glance; imagine a user interacting with a web browser on a wall screen. Is the web browser operating on behalf of the user, or is it operating on behalf of the room? The answer bears heavily on the behavior exhibited when another person enters and tries to view a web page – if the existing browser is working on behalf of the user, a new window should be opened, otherwise the existing page should be shared.

Once agents are divided into societies, we encode information about the real world entities into the ambassadors, so that the agents can utilize the information during operation. This can include configuration information requested by the user or space, resource management subsystems, and connections between the different societies. For example, even if the web browser agent above is operating on behalf of the room, it might still need to keep track of which user requested its current data. To accomplish this, the ambassador agent for the room can track the current users and the connections between them.

The result of this is that the details of a society become abstracted away behind the ambassadors, and agents within a society can be seen merely as the capabilities that the society presents. This makes it far easier for a user moving from one environment to another to transfer the display functions of his or her work (web pages being browsed, email agents) to another space, without being bogged down in the details of the environments. Indeed, most of the details can be encapsulated into a resource management framework (Gajos, 2000).

Mobile Devices

Agents that control devices are usually easy to assign when operating in the context of an intelligent room – a display controller agent is clearly associated with one of the room’s displays, for example. However, some devices seem to be more closely associated with the user; laptop computers, pagers, and PDAs are good examples.

Assigning these devices to the room where they are currently located raises problems when the user moves. This would require the ability to transfer an agent from one society to another, updating all references to that agent. Unfortunately, this is not currently implemented in the current agent infrastructure.

Assigning agents for these devices to the user's society raises other issues. Although a room's devices are constantly present, the user is not attached to his pager or laptop, so an agent controlling the device cannot assume that its outputs will actually reach the user. In addition, this breaks the abstraction barrier mentioned in the previous section, making agent implementations far more complex.

We are solving these issues by creating a set of mobile device societies, or "mobile spaces," which can handle these devices. This both preserves the abstraction between users and the devices that they are attempting to use, and makes it possible to decouple the mobile devices from the user as well as the spaces they inhabit. Users can carry the devices with them, leave them behind, use them as extra displays or input devices, in as flexible a fashion as the resource management system allows.

This has two implications on the system. First, in order to represent one of these "mobile spaces" as being in a room, we need to encapsulate the relationship that a space is within another. Second, the user needs to have the ability to be present in multiple spaces at once. In terms of infrastructure, this is straightforward, as the user's ambassador can simply maintain a list of spaces associated with the user. However, it does compound the problem with resource management, since the user's ambassador now needs to coordinate with several device societies in order to utilize devices.

Managing Groups

However, there is another class of agents which cannot clearly be assigned to either a user or a society. When dealing with collaborative applications, there will likely be agents operating in each individual user's society, but there usually need to be coordinating agents managing the collaboration (Ellis & Wainer, 1999).

Consider an agent-based shared whiteboard system. Although each user can have his own interface for displaying the shared space and relaying user inputs, the current state of the whiteboard needs to be stored in a separate agent, which manages the interactions with the different clients.

Given that users are working in separate spaces, and could move during the session, the coordinator agent shouldn't exist in one of the spatial societies. One possibility is to have the whiteboard clients elect a representative user to hold the coordinator, but if that user needs to leave the session, the coordination will become unstable.

To handle these situations, we create temporary *ad hoc* societies which are automatically created by the first client attempting to use such a groupware system, and which will exist solely for managing these collaborations. The ambassador for these "group societies" manages the interactions, and automatically destroys the societal information when the last client disconnects.

Current Status

As this is an ongoing project, some issues involved in the ambassador infrastructure are still being resolved. Initial tests integrating simplified ambassadors and simple resource management systems show promise, in that it does enable users to move state easily between environments. Work is currently proceeding on handling group societies and nested spaces.

References

Bobick, A., Intille, S., Davis, J., Baird, F., Pinhanez, C., Campbell, L., Ivanov, Y., Schütte, A., & Wilson, A. (1998). Design decisions for interactive environments: Evaluating the kidsroom. In (Coen, 1998), 7–16.

Brumitt, B., Meyers, B., Krumm, J., Kern, A., & Shafer, S. (2000). Easyliving: Technologies for intelligent environments. *Handheld and Ubiquitous Computing: Second International Syposium* (pp. 12–29). Bristol, UK: Springer.

Coen, M. (Ed.). (1998). *Proceedings of the 1998 AAAI spring symposium on intelligent environments*. American Association for Artificial Intelligence. Menlo Park, California: AAAI Press.

Coen, M., Phillips, B., Warshawsky, N., Weisman, L., Peters, S., Gajos, K., & Finin, P. (1999). Meeting the computational needs of intelligent environments: The metaglue system. In (Nixon et al., 1999), 201–212.

Ellis, C., & Wainer, J. (1999). *Multiagent systems: A modern approach to distributed artificial intelligence*, chapter Groupware and Computer Supported Cooperative Work, 426–452. Cambridge, MA, USA: MIT Press.

Gajos, K. (2000). A resource management system for the intelligent room. Master's thesis, Massachusetts Institute of Technology, AI Lab, Cambridge, MA.

Nixon, P., Lacey, G., & Dobson, S. (Eds.). (1999). *Managing interactions in smart environments: 1st international workshop on managing interactions in smart environments (manse '99)*. London: Springer-Verlag.