Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science

Proposal for Thesis Research in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

TITLE: Moving Towards Multi-User, Multi-Spatial Collaborative Intelligent Environments

SUBMITTED BY: Stephen Peters
50-1 Garden St.
Cambridge, MA, 02138

(SIGNATURE OF AUTHOR)

DATE OF SUBMISSION: February 16, 2001

EXPECTED DATE OF COMPLETION: Fall 2002

LABORATORY WHERE THESIS WILL BE DONE: MIT Artificial Intelligence Laboratory

BRIEF STATEMENT OF THE PROBLEM:

Most of the currently available intelligent environments are targeted towards a single user operating a single environment. However, as technology progresses, it is reasonable to expect that such environments will need to handle the presence of several users, and need to allow users to travel from one environment to another, and adapt the users' needs to the spaces they are in. This thesis attempts to show that adding straightforward knowledge representations of users and spaces to an existing intelligent environment system will enable these types of interactions.

Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts 02139

Doctoral Thesis Supervision Agreement

TO: Department Graduate Committee
FROM: Howard Shrobe, Principal Research Scientist, MIT AI Lab

The program outlined in the proposal

TITLE: Moving Towards Multi-User, Multi-Spatial Collaborative Intelligent Environments
AUTHOR: Stephen Peters
DATE: February 16, 2001

is adequate for a Doctoral thesis. I believe that appropriate readers for this thesis would be:

READER1: Randall Davis, Professor of EECS
READER 2: Trevor Darrell, Assistant Professor, EECS

The facilities and support required are available, and I am willing to supervise the research and evaluate the thesis report.

Howard Shrobe, Principal Research Scientist, MIT AI Lab ______ date

COMMENTS:

Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts 02139

Doctoral Thesis Reader Agreement

TO:	Department Graduate Committee
FROM:	Randall Davis, Professor of EECS

The program outlined in the proposal

TITLE:	Moving Towards Multi-User, Multi-Spatial Collaborative Intelligent Environments
AUTHOR:	Stephen Peters
DATE:	February 16, 2001
SUPERVISOR:	Howard Shrobe, Principal Research Scientist, MIT AI Lab
OTHER READER:	Trevor Darrell, Assistant Professor, EECS

is adequate for a Doctoral thesis and I am willing to aid in guiding the research and evaluating the thesis report as a reader.

Randall Davis, Professor of EECS ______ date ______

COMMENTS:

Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts 02139

Doctoral Thesis Reader Agreement

TO: Department Graduate Committee
FROM: Trevor Darrell, Assistant Professor, EECS

The program outlined in the proposal

TITLE:	Moving Towards Multi-User, Multi-Spatial Collaborative Intelligent Environments
AUTHOR:	Stephen Peters
DATE:	February 16, 2001
SUPERVISOR:	Howard Shrobe, Principal Research Scientist, MIT AI Lab
OTHER READER:	Randall Davis, Professor of EECS

is adequate for a Doctoral thesis and I am willing to aid in guiding the research and evaluating the thesis report as a reader.

______________________________ ____________
Trevor Darrell, Assistant Professor, EECS date

COMMENTS:

1 Background

Recent research into intelligent environments ([NLD99], [Coe98]) 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 spaces usually limit themselves to a single intelligent environment. As we move towards a world where people move between intelligent spaces ([Der99]), it becomes necessary 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.

Other research projects have tried to handle some of these issues without really representing individuals as such. Bobick et al. in their KidsRoom ([BID+98]) project could track many different children, but did not try to deal with them as individuals – instead, it performed its actions based on the entire group's behavior. Stiefelhagen et al. ([SYW99]) could focus in on the current speaker within a meeting environment, but did not try to assign information to the individuals involved beyond the simple calculations of gaze direction.

The Metaglue system developed by Michael Coen et al. at the Massachusetts Institute of Technology Artificial Intelligence Laboratory ([CPW+99]) has been in active development over the past few years, and has evolved into a fairly comprehensive multimodal system. It also, however, is still tied to both one user interacting with a single space.

The EasyLiving project[BMK+00] 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.

2 Intentions

This thesis will extend an existing single-user, uni-spatial intelligent agent-based environment by adding a small knowledge representation, and show that this creates a reasonable multi-spatial, multi-user environment. This work endeavors to show that such a limited knowledge representation, encompassing users, spaces, and notations about the roles of each, can exhibit behavior that responds very well to the user's actions, and creates a substrate for dealing with collaboration among the spaces and users.

The knowledge representation does not need to be extensive in order to be useful for these tasks. An explicit separation of users and spaces, coupled with information about capabilities and roles is all that is needed to make the system work cleanly:

- Many of the existing intelligent environments use a device-centric focus, so a user effectively is "whoever is speaking into the microphone", and a space is merely the currently active collection of devices. In order to handle a collection of users or spaces, we need to provide the system with a notion of what these entities are.

- Where many speech systems are designed for environments with a single user providing voice input either through a single microphone or over a telephone, the system needs to be redesigned to separate out multiple microphone inputs. Providing a simple model of a room and the input devices is a straightforward first step for this task. In addition, the existing vision inputs for the room are being reengineered to perform face recognition, and even to identify the current speaker and steer arrays of microphones towards his location.

- Adding in the representational information will also assist in generating systems that can better adapt to what the user is trying to do. Systems that try to "learn" the behavior of people can only adapt to many different users if there is a clear definition of what a user is. Similarly, being able to have a user's desires follow him into a different space requires that the space be carefully represented inside the system, and that information on the space's capabilities be provided.
- Useful collaboration technologies, in which different members of a group can work together on a single project, requires some straightforward knowledge of roles for users and groups.

What the knowledge representation will contain, then, is a compact representation of users, spaces, and groups within the existing agent infrastructure, coupled with a small information base describing capabilities for spaces and roles for users.

Further projects could also involve adding device management ([Gaj00]), so that agents can make requests for devices easily and even seamlessly transfer their output from one display to another, or integrating information management ([Bro01]), so that complex data can be passed among the various users and devices and displayed in whatever way makes the most sense.

3 Approach

The approach taken here will be to take the agent system's representation of a collection of agents (a "society", to use the nomenclature suggested by Coen), and split it into three different types: societies that represent spaces, societies that represent people, and societies that represent groups of other societies. This encodes the notion of agents that are operating on behalf of or in concert with the entity being represented. For each of these societies, there is a `Society` agent, which will contain information about the represented entity.

Of course, with some agents this brings up a central question – how do you decide whether an agent is operating directly on behalf of a person or whether it is acting on behalf of the space. This is especially true with display agents where are displaying information for a user on a display that is in the space. We solve this in part by separating out agents that control a device from agents that are actually performing the display.

3.1 Designing a Running System

One way to look at this is to look at a running system, and examine where the different agents really should lie. For example, let's take a user (named Alice) and place her in a conference room (called "room"). She'll be operating a map system, which also interfaces to a data-gathering and query system known as START ([Kat90]). Devices in the room include two projectors, a microphone for speech input and a set of speakers for device output.

There are two main societies here – the *alice* society representing the user and the *room* society representing the conference room itself. Each of these has a `Society` agent handling information for the entity (see Figure 1).

There is a subset of agents which are definitely associated with the room – the agents that control and query the room's devices directly. These would include `Speech` controllers that handle and dispatch Alice's speech, `DisplayManager` controllers which deal with the projector displays and can allocate the display use for programs, and a `SpeechOut` agent which can make voice announcements or queries to whichever user is in the room.

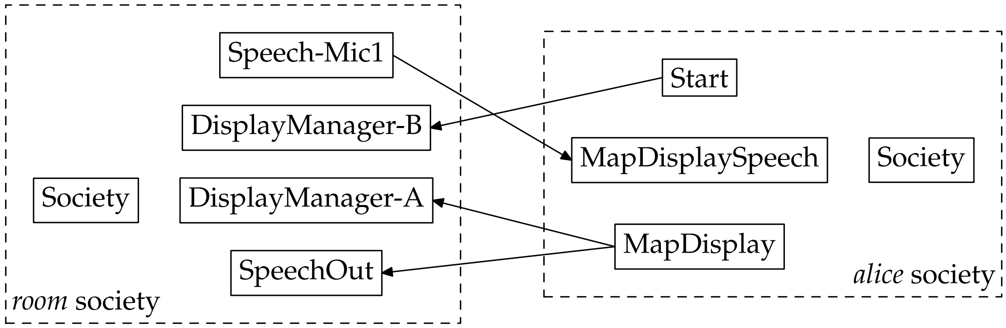

Figure 1: Agent layout for a simplified set of two societies

Since we separate out the controllers from the actual display of information, the agents that are actually performing the display – the `MapDisplay` agent and perhaps the `Start` agent as well – can live in Alice's society, neatly indicating that they are operating on Alice's behalf. Alice can also keep control over the `MapDisplaySpeech` application that handles speech events from whatever space she is in.

This separation means that Alice can travel easily into another space that has a different device setup, and her agents can simply disconnect from the agents in *room* and set up new connections to the associated agents in the new space.

3.2 Designing the Knowledge Base

Once the societal division is accomplished, more work still needs to be done in fleshing out the components of a society, especially in determining what information the different types of `Society` agents need to track in order to function properly. In addition, the "on-behalf" relationship can't always be modeled totally through the societal membership function, so more work needs to be done towards letting agents detail that relationship explicitly.

However, the other main component of the knowledge base is the knowledge of users and spaces needed to allow for collaboration and user movement. This will be provided for by specifying different capabilities that the users and spaces can handle, and the different roles that they perform.

For example, the *room* society has several capabilities, defined by the different devices that it deals with:

```
(has-mic room Speech-Mic1)
(has-speech room SpeechOut)
(has-display room DisplayManager-A)
(has-display room DisplayManager-B)
```

As seen, if there are multiple displays available, this can be handled through the use of multiple capability listings. When Alice moves to a new environment, her agents can simply use the new environments capabilities to decide how best to present information.

Similarly, Alice's society can describe a number of roles that she can perform. These can be as mundane as describing her membership in groups or can give the system information about her state:

```
(is-grad-student alice lab)
(is-programmer alice lisp)
(is-located alice room)
```

More work needs to be done in honing the types of information that the system will need to handle. The knowledge base doesn't need to be limited to that subset, but that subset needs to be described and published to agent programmers.

3.3 Building the System

Once the layout of the infrastructure is complete, the next step will be to write the necessary agents into the agent system and augment the system to handle the `Society` agents. This will also entail adding code to deal with users as they move from room to room, and being able to have multiple users sharing not only the same room, but potentially also the same device within the room.

Once that is in place, work will progress on making agents that can use the capability and role information to act as secretaries for users and rooms – managing schedules and dispatching messages to users based on their knowledge of where the user is located.

3.4 Reworking the Perception Model

As mentioned before, the existing agent system has a device-centric focus, and thus handles perception events in an *ad hoc* manner – camera controllers watch for changes in the room state, make assumptions about what is occurring, and update agents based on those assumptions. Unfortunately, this model leaves little room for coalescing several events into a higher-level understanding of what is occurring in the space. In addition, the assumptions made when dealing with just one person in the space mean that multiple people are either ignored or just cause confusion.

In order to deal with multiple people, the system's handling of events must be reworked so that perception events can be assigned to the user who triggered them. Since the system will also need to be able to identify and recognize people, it is likely that several different sensors (cameras, microphones, pressure pads) will combine to create the appropriate identification. Therefore each space will require a new method of handling and coalescing events.

One way of doing this is to have the `Society` agent handling the space also maintain an "event blackboard", where the different perceptual agents can write information about the events they're noticing, and other agents can read the information and generate new events from the constituent pieces. Some of the initial work for this is described by Coen in an upcoming paper([CW00]), although he is mostly concerned with the training problems for the sensors and grouping devices into spatial areas.

Each event written onto the blackboard can describe the events as it saw them, and mark them with a timestamp and confidence level. For example, a camera watching for entrances and exits through a doorway could write an event tuple of the form

```
(event exit DoorCamera-1 15:43:20.18 20%)
```

to the blackboard to indicate that it believes that a person has exited the room, but has low confidence in that diagnosis. If there are further devices monitoring the door, they might also put in similar information.

Other agents can monitor the blackboard and try to combine the several "exit" events into a single exit event with a far higher confidence. In addition, agents can monitor for low-confidence events and steer other sensors towards the area to perform confirmation of the event and gain more confidence.

Given the coalescing of events, the agent system can be used to try to organize a series of perceptual events into descriptions of event series – "Joe walked in and took something from the bookshelf", for example. By getting good confidence matches of the entrance event, a 'bookshelf event', and the identification of Joe, you can coalesce the events into high-level understanding that can be used by the agent system to communicate with the user about high-level behavior.

One possible implementation would be to integrate the blackboard into a rule-based expert system grafted into the agent framework. The agents which wish to coalesce events can specify new rules which take patterns for events and assert new events into the system.

3.5 Extensions for Groups and Portable Spaces

The third type of society, handling groups of people or spaces, needs to be created so that requests to a group of people can be efficiently passed on to all the group's members at once. For example, sending an announcement to a group's `AnnouncementHandler` agent should send a copy of the announcement to the `AnnouncementHandler` for each of the group members, where presumably it can decide to notify the user through whatever means the user prefers – whether it be through email, a voice input, or a display on the LED sign.

Note as well that the idea of a "space" does not have to be, and indeed should not be, relegated to rooms that have a large collection of devices. To create a system that is truly omnipresent, we should be able to allow the user to carry with him devices that can be used by the system and notified through the agent mechanism. For example, if a user carries a pager it would make sense to define a "portable space" that can be manipulated through the `PagerOutput` agent. Similarly, a laptop can also be a mobile space which can contain many of the input and output devices of a full room, although without as much versatility.

4 Evaluation Criteria

The main criteria for success will be to perform tasks that take advantage of the new capabilities of the infrastructure. Several of these tasks are outlined in the following sections.

4.1 State Transfer Among Spaces

In order to correctly handle the differing device lists for multiple rooms, the system should be able to easily transfer a user's state to another intelligent space. One easy way to accomplish this is to ensure that an agent running in a user's space can adjust its output location when informed about the user's entrance into a new space.

The scenario we'll be aiming for is allowing the user to work with speech input agents and web output agents in one room, walk into the new conference room, and have the output immediately transfer to one of the conference room displays. Microphone input will also be switched over at the same time.[1]

Since the notion of a space can also be applied to portable devices, this demonstration might be best exemplified by a transfer of an agent's display from a laptop screen to a projector upon entering the conference room.

[1] At the current level of technology, this will require that the user switch microphones upon entering the new room, unless we have some way of changing the frequency of the new wireless microphones.

4.2 Organizing Multiple People

The system needs to demonstrate that it can correctly separate information for different users and handle them according to their preferences.

To demonstrate this, we need to have a way to identify to a space that there are multiple people inside and give their identities. Once those identifications are made (possibly through keyboard or voice input), the system should be able to partition devices to the individuals in the room without confusing them. One example will be to use multiple web display agents on separate projectors, and demonstrate that one user's request to update the display only effects one of the displays.

4.3 Collaboration Behavior

The group society behavior also needs to be demonstrated. This can be done with a straightforward group society agent that will serve as the front end to a group of people, and, as described in section 3.5, forwards announcements to many individual handler agents.

None of the above scenarios should be tied to an individual room or people. The entire point of this work is to create agents that are truly acting on behalf of a user or a space. It should be just as straightforward to demonstrate moving from a laptop to a conference room as it is to go in the opposite direction.

5 Work Outline

The current plan for designing, implementing and testing the system is as follows:

May 2001	Finalize implementation of initial society agents and deploy in existing room. Implementation should contain integration with resource managers if they exist. Create several test users.
September 2001	Implement and demonstrate ability to move running state between two separate intelligent environments. This will likely require identifying users by hand.
February 2002	Demonstrate ability to handle multiple users operating in a single conference room.
May 2002	Implement and demonstrate group behavior.
September 2002	Finish demonstrations and write up thesis.

References

[BID+98] Aaron Bobick, Stephen Intille, Jim Davis, Freedom Baird, Claudio Pinhanez, Lee Campbell, Yuri Ivanov, Arjan Schütte, and Andrew Wilson. Design decisions for interactive environments: Evaluating the KidsRoom. In Coen [Coe98], pages 7–16.

[BMK+00] Barry Brumitt, Brian Meyers, John Krumm, Amanda Kern, and Steven Shafer. EasyLiving: Technologies for intelligent environments. In Peter Thomas and Hans W. Gellersen, editors, *Handheld and Ubiquitous Computing: Second International Syposium*, pages 12–29, Bristol, UK, September 2000. HUC, Springer.

[Bro01] Ruben E. Brown. Information management in intelligent environments. Master's thesis, Massachusetts Institute of Technology, AI Lab, Cambridge, MA, 2001.

[Coe98] Michael Coen, editor. *Proceedings of the 1998 AAAI Spring Symposium on Intelligent Environments.*, Menlo Park, California, 1998. American Association for Artificial Intelligence, AAAI Press.

[CPW+99] Michael Coen, Brenton Phillips, Nimrod Warshawsky, Luke Weisman, Stephen Peters, and Peter Finin. Meeting the computational needs of intelligent environments: The Metaglue system. In Nixon et al. [NLD99], pages 201–212.

[CW00] Michael Coen and Kevin Wilson. Learning spatial event models from multiple-camera perspectives in an intelligent room. In submission, 2000.

[Der99] Michael L. Dertouzos. The future of computing. *Scientific American*, July 1999.

[Gaj00] Krzysztof Gajos. A resource management system for the Intelligent Room. Master's thesis, Massachusetts Institute of Technology, AI Lab, Cambridge, MA, 2000.

[Kat90] Boris Katz. Using English for indexing and retrieving. In P.H. Winston and S.A. Shellard, editors, *Artificial Intelligence at MIT: Expanding Frontiers*, volume 1. MIT Press, Cambridge, MA, 1990.

[NLD99] Paddy Nixon, Gerard Lacey, and Simon Dobson, editors. *Managing Interactions in Smart Environments: 1st International Workshop on Managing Interactions in Smart Environments (MANSE '99)*. Springer-Verlag, London, 1999.

[SYW99] Rainer Stiefelhagen, Jie Yang, and Alex Waibel. Modelling focus of attention for meeting indexing, 1999. ACM Multimedia '99, Orlando, Florida, Oct. '99.