# Activity Zones for Context-Aware Computing

Kimberle Koile[1], Konrad Tollmar[2], David Demirdjian[2],
Howard Shrobe[1], and Trevor Darrell[2]

[1] Agent-Based Intelligent Reactive Environments Group
[2] Visual Interface Group

MIT Computer Science and Artificial Intelligence Laboratory
Cambridge, MA 02139
{kkoile, konrad, demirdji, hes, trevor}@ai.mit.edu

**Abstract.** Location is a primary cue in many context-aware computing systems, and is often represented as a global coordinate, room number, or a set of Euclidean distances to various landmarks. A user's concept of location, however, is often defined in terms of regions in which similar activities occur. We discuss the concept of such regions, which we call activity zones, and suggest that such zones can be used to trigger application actions, retrieve information based on previous context, and present information to users. We show how to semi-automatically partition a space into activity zones based on patterns of observed user location and motion. We describe our system and two implemented example applications whose behavior is controlled by users' entry, exit, and presence in the zones.

## 1 Introduction

The utility of an application can be increased greatly by taking into account the specific context in which the application is used. Users, however, do not like the burden of explicitly stating context information. The implicit control of applications using passively sensed context cues frees users from the burden of explicitly specifying service details, e.g., for whom a service should be performed, where it should occur, and how it should be delivered. Location is one of the primary context cues in context-aware systems, e.g., [6]. Many of these systems define location in geometric terms such as a 2D or 3D position coordinate, room identifier, or a set of distances to known landmarks. While these definitions are useful, they neglect how people actually use physical space.

One way to understand the use of physical space is in terms of zones that are created by elements of physical form and that support human activity, e.g., [21, 25]. Architects, for example, design eating places, gathering places, and places of repose, e.g. [3, 24]. Anthropologists and sociologists talk of "hidden zones" in offices [17]. CSCW researchers interested in home environments have studied public and private zones for use of communication media [27], and have identified activity centers where particular tasks take place [10]. In workspaces, we think of zones for quiet, solitary work; zones for informal meetings; zones for formal presentations [14, 34]. A table and accompanying chairs create a zone in which we might meet with colleagues. A whiteboard creates a zone in front of it; people draw on the whiteboard while standing in this "whiteboard zone".

We make two key observations about identifying such zones. Thinking bottom up, we can identify zones by observing people as they go about their daily activities, partitioning a space based on people's locations and motions. Thinking top down, we can define a taxonomy of zones based on prototypical human activity and physical form that supports that activity. Such zones, identified using either or both of these approaches, can be thought of as representing regions of similar context. With such activity-dependent zones, one can build more useful context-aware computing applications by (1) identifying meaningful zones to which users can attach semantics and preferred application behavior, e.g., behavior that would be triggered upon entry, exit or presence in a zone; and (2) enabling inference of human activity at various levels of abstraction, e.g., three people sitting in chairs around a table, vs three people having a meeting.

In this paper we describe our implementation of a system that identifies "activity zones" semi-automatically using techniques from computer vision and artificial intelligence. We present an implemented experimental system in which transitions between zones successfully control device and application behavior.

We begin by reviewing previous work, then discuss the concept of activity zones, our experimental system, and ongoing and future research efforts.

## 2 Previous Work

The study of context and its role in ubiquitous computing systems is an active research field, with many definitions for context and context-awareness, e.g., [12, 13, 32]. Central to the notion of context is location, since many applications and services are conditioned on the place where they should be performed or displayed. Location-aware computing has become a topic of active research, and many schemes have been proposed for providing location cues using IR, RF, ultrasound, and computer vision tracking systems. (For a survey, see [22].)

Early systems employed ad-hoc representations of location, usually tailored to specific sensor data representations. Recently a general scheme for device independent location representation and sensor fusion has been proposed, using a layered abstraction model based on proximity and position measurements [23]. The majority of location-awareness schemes report raw 2D or 3D position information, room identity, and/or proximity to a beacon. These cues are useful for many tasks, but they are indifferent to the physical form or use of the space, and thus are insufficient in some cases. A person may be equally close to a whiteboard or a table, for example. If a display system knows that the person is moving back and forth in a standing position, it can infer that the person is at the whiteboard and that a nearby wall is a better display location than a computer monitor on the table.

A few systems for location awareness are able to report information about sub-regions or furniture, and/or adapt over time based on observed behavior. The Easy-Living system used a map of regions to indicate places in the environment associated with specific context features, usually furniture [7]. These maps were drawn manually and provided to the system, rather than being learned from observed behavior. Similarly the Sentient Computing System's notion of spatial containment allows bounds for 2D regions associated with positions of active devices [2], e.g., an oval area in front of a computer display. A system for automatically mapping an environment based on the movement of personnel in a location-aware environment was described in [19]. This system was adaptive and learned from observing user behavior, but formed a map of a large scale environment and did not find contextually relevant regions within rooms.

In our current research, we represent context by means of regions that we call activity zones, which are identified by observing users' locations and motions. The term "activity zones" is used in [30] to describe spatial distribution of activities in an office setting. The zones represent regions within a room, as ours do, but are computed by analyzing a large corpus of camera images rather than by means of real-time tracking software. In addition, the zones are used to inform the design of new office layouts rather than in ubiquitous computing applications.

In contrast to previous work, our activity zone representation is both fine-grained (i.e., smaller than a room) and learned from observing patterns of user behavior. It supports applications that can make productive use of inferences about a user's activity beyond simple position and proximity cues, without requiring the user to draw a map. We develop our scheme in an interactive framework: the system learns the geometry of the activity zones automatically from observed behavior, and relies on the user to associate semantics or rules with events related to those zones.

## 3 Activity Zones

Physical form—e.g., walls, furniture—partitions space into zones that are places of human activity. Walls create a zone in which people might play music; furniture creates a zone in which people might read or talk. In the floor plan shown in Fig. 1, there is a zone created by the doorway, a zone created by the sofa, and a zone created by the corner desk, table, and chair.

Zones defined by physical form are useful as representations of context, but only partially capture a person's context—they ignore a person's use of a space. Rather than building a model of a space and its furniture, as for example in [8], we can partition a space into zones by observing human activities. We call these partitions activity zones; e.g., the area in which a group of people are standing in a hall talking. If we take "observing human activity" to mean recording people's locations and motions, then for a floor plan such as the one shown in Fig. 1, we would ideally find the zones shown in Fig. 2: zone 1 is a region in which people stand, zones 2 and 3 are regions in which people sit, zone 4 is a region in which people walk. Note that zones 1 through 3 correspond to the physical zones shown in Fig. 1. They contain extra information, however—whether the person sits or stands in these zones. Zone 4 corresponds to an access corridor used for walking between the three other zones. To identify such a zone in a model that represents only the physical space, such as that shown in Fig. 1, one would have to represent circulation paths explicitly.

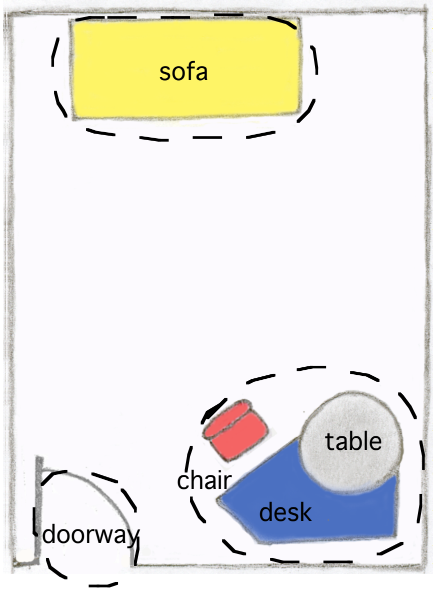

**Fig. 1. Three zones created by physical form**

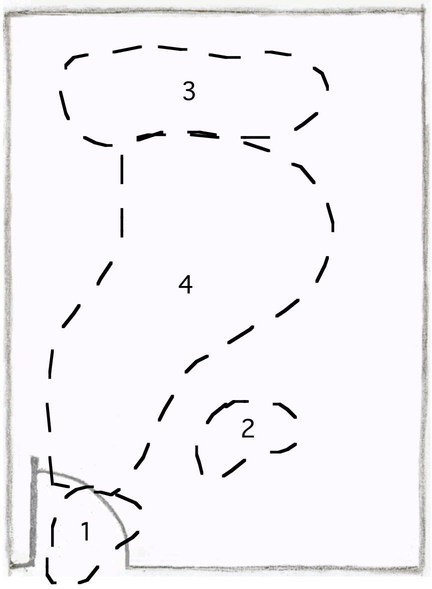

**Fig. 2. Four zones identified by observing human activities**

Both physical zones, shown in Fig. 1, and activity zones, shown in Fig. 2, can be thought of as representing regions of similar context. Physical zones represent location contexts; activity zones represent location and motion contexts. Physical zones could be inferred from observation of static furniture configuration, but activity zones need to be learned from statistics of human behavior. Application behaviors can be controlled by a person's entry, exit, or presence in either type of zone. Activity zones, with the extra information about user motion, enable application behaviors to be tied more closely to what people are doing rather than just where they are.

To construct activity zones, a tracking system observes people's activities over time, say a week, and constructs a map of activity zones. A user, or eventually a machine learning program, then may attach preferred application behavior to activity zones. A user also might attach semantics to the activity zones so that application behaviors could be specified for types of zones instead of individual zones. If a user, for example, did not want to receive phone calls while reading, she might label a particular zone as a reading zone, and indicate that calls should be held while she is reading. Once preferred behavior has been specified, the tracking system posts events about people's entry, exit, or presence in particular zones. An accompanying notification system informs interested applications of the events, and the applications react accordingly.

In this description of activity zones, the map is static—it is created by observing people in a space, then used assuming that the arrangement of space does not change. Yet zones are often correlated with furniture location, and in today's workspaces furniture is often moved. What happens, then, when a sitting zone no longer contains the chair it once did? One could map furniture locations to zone locations when an activity zone map is created, then periodically check that the current furniture locations match those in the current zone map. We discuss the issue of dynamic activity zones further in Sect. 7.

## 4 Scenarios

The following scenarios illustrate the activity zones concept. Consider a workspace containing zones such as those shown in Figs. 3 and 4. Note that the zone near the round moveable table in Fig. 4 is what we have called a dynamic activity zone.

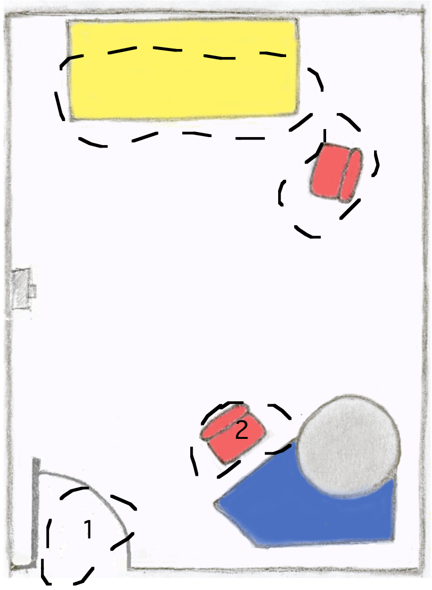

**Fig. 3.** **Consider two zones, one at doorway one at corner desk chair; projector is shown at left**

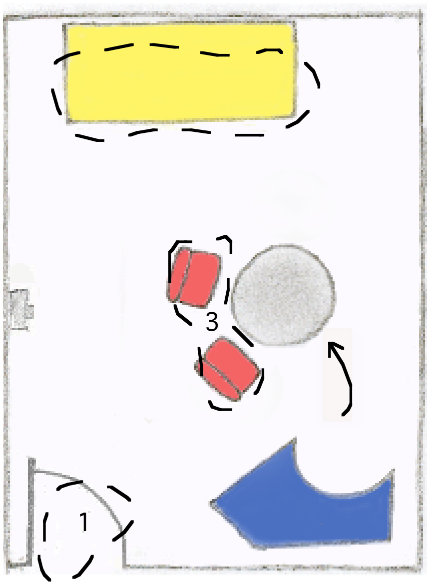

**Fig. 4.** **New zone is created when round table is moved**

Scenario A: Jane walks into her office and the overhead lights and the teapot turn on; the newspaper headlines are displayed on the wall near her desk. The room greets her and asks if she would like to be reminded of her calendar events for the day. She says yes, sits down at her desk, and her calendar is displayed on the computer screen at her desk. She notices that a student, Lauren, is coming by soon for a meeting about a term project. Jane asks the room to retrieve the notes from her last meeting with Lauren and to project them when she and Lauren start their meeting.

Scenario B: Lauren arrives, Jane greets her and invites her to come in. Jane moves the table into the middle of the room and invites Lauren to sit down at the table. As Lauren sits down, the notes about her project are projected on the wall between the table and desk. Jane and Lauren start to discuss the project.

Scenario C: After discussing the project, Lauren leaves. As Jane moves the table back to its original location, the projector turns off. Back at her desk, Jane notices that the message light on her phone is lit and asks the room to play her phone messages.

In Scenario A, several preferred behaviors were triggered upon entry into the doorway zone: “turn on overhead lights”, “display news headlines”, “turn on teapot”, “ask about showing calendar”. Jane’s entry into the zone near the desk triggered display of the calendar on the display device appropriate for that zone, the computer screen on the desk. Scenario A also illustrates the creation of a dynamic event trigger: Jane requests that the room do something when a future event occurs, namely that it display notes when her meeting starts. Note that in order to identify a meeting, the room would need additional knowledge. It would need to know, for example, that meetings happen when more than one person is in zone 3, or that meetings happen at tables and there is a table in zone 3. Without this additional knowledge, Jane would have had to request that the room display the information when she and Lauren entered zone 3.

Scenario B illustrates the display of information in an appropriate place based on the meeting starting in zone 3. Simple proximity to devices would not have worked in this example, because Jane and Lauren were equidistant between the projector and the computer display on the corner desk.

Scenario C illustrates that phone calls are held during meetings. It also shows the projector turning off when Jane moves the table back to its original location.

Together these scenarios illustrate the use of context to trigger room actions automatically, to retrieve information, and to present that information to users. These uses of context are similar to those discussed in [8] and [13].

## 5 Implementation

We have implemented an activity zone system and two of the context-aware application behaviors mentioned in the above scenarios: device control and selection of display location and method. Our activity zone system is part of a larger system that provides services in an intelligent environment. It embodies a perceive-reason-act paradigm, as illustrated in Fig. 5, and is organized using a blackboard architecture [15]. Perceptual systems, such as the person tracker, post events to a blackboard. The blackboard provides a reasoning system with a shared memory that represents the current state of the world—e.g., the activity zone map in use; pending and processed events; the people in the space and their contexts, represented as motions and entry, exit, or presence in particular zones. Knowledge sources associated with the blackboard do forward inference, abstracting events into higher level context statements (e.g., a meeting in zone 3) or mapping events to requests for action (e.g., turn on the lights). The requests for action are sent to device controllers and other applications, which in turn process the requests.

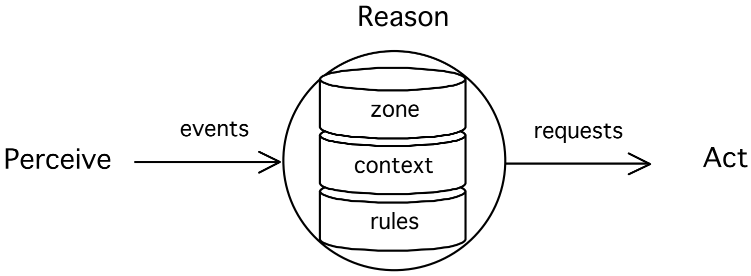

**Fig. 5. Overview of system architecture**

We have installed our system in a workspace similar to the one described in the above scenarios, focusing on furniture that can be configured easily to create zones for individual work and collaborative work. Fig. 6 is a sketch of the floor plan. Fig. 7 is a photograph of the space.

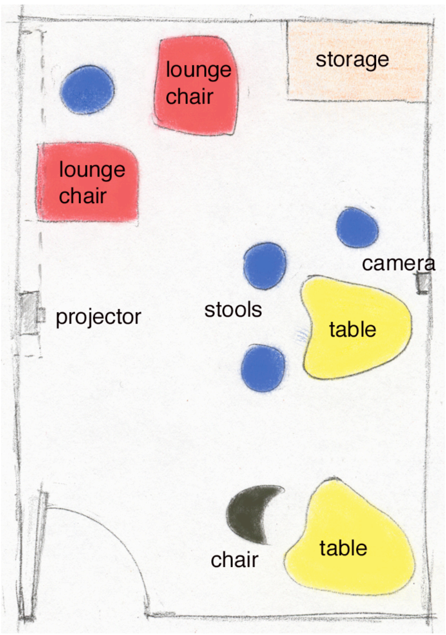

**Fig. 6. Floor plan sketch of workspace**

**Fig. 7. View of workspace from doorway**

### 5.1 Tracking System

To support activity zones, a person tracking system must provide information in real-time on the number of people in a space and each person's location and motion, e.g., represented by height and velocity. These are general requirements, and many different tracking approaches can be used, including those that track wearable or handheld devices using IR, RF, infrared, GPS, 802.11 [22]. We have chosen to use a passive person tracking system based on computer vision because it does not require any additional devices to be worn or carried by users. The concept of activity zones and our techniques for constructing and manipulating zones, however, are independent of the tracking system used, as long as the tracker provides the information specified above.

We use a multi-camera stereo-based tracking system to track people in indoor environments. (More details about the tracking system can be found in [11].) The tracker provides a history of 3D information for every person in the observed space. The information is a triple $(x, y, h)$, where $x, y$ are the coordinates of the person in the ground plane, and $h$ is the height of the top of her head above the floor. Since tracking data are time-stamped, the instantaneous velocity $(v_x, v_y, v_h)$ can be derived. We then characterize a person at location $(x, y)$ using the activity feature $f(x,y) = (h, v, v_{lt})$, where $h$ is the height, $v$ is the instantaneous ground plane velocity norm, and $v_{lt}$ is the average ground plane velocity norm over a certain period of time. By using the activity feature $f(x, y)$, we can capture the configuration (sitting, standing) and movement of a person over both short and long periods of time.

To estimate an activity zone map, we track people in a space for a period of time, collecting a dense set of activity features $f(x, y)$ and locations $(x, y)$, then segment the activity features using a two-step clustering process. We first cluster the activity features into classes, each representing a similar activity (i.e., configuration and movement), which is represented as an average activity feature $F_k$. Then for each class, we cluster the associated $(x, y)$ locations into regions. The resulting regions represent activity zones, regions in 2D space that are characterized by an average activity $F_k$. As different activities may happen at the same location, activity zones may overlap. Once an activity zone map has been created, a person's entry, exit, or presence in an activity zone is identified by matching the person's instantaneous location and activity features to the map. (For more details of the clustering and matching algorithms, see [11].)

The length of time for collecting data for an activity zone map is an open research question; we typically track people for a day. People can be detected with an accuracy of about 20 to 30 centimeters. We have run experiments with the system successfully tracking 8 people in the space at the same time.

The figure below shows a portion of an activity map for our workspace. The map is represented using simple XML primitives in order to allow heterogeneous agents, such as one that provides a graphical user interface for visualizing and labeling zones, to easily access and manipulate the map data. In this example, the map contains three zones numbered 0, 1, 2 and labeled "desk", "table", and " lounge". The numbers are supplied by the tracker; labels are added by the user via a simple graphical user interface. Displays from that interface, which we call the zone editor, are shown in Figs. 9 and 10.

```
<?xml version="1.0"?>
 <amap xsize=200 ysize=200 im="835-PTGO.jpg">

   <zone id=0 label="desk" height=1.1 velocity=0.1 color="ff0000">
        12,182 12,184 38,188 …
   </zone>

   <zone id=1 label="table" height=1.7 velocity=0.5 color="00ff00">
        100,182 100,180 89,154 …
   </zone>

   <zone id=2 label="lounge" height=0.8 velocity=0.05 color="0000ff">
      150,130 150,132 135,118 …
   </zone>

  </amap>
```

**Fig. 8. Activity-map in XML. Labels are user supplied**

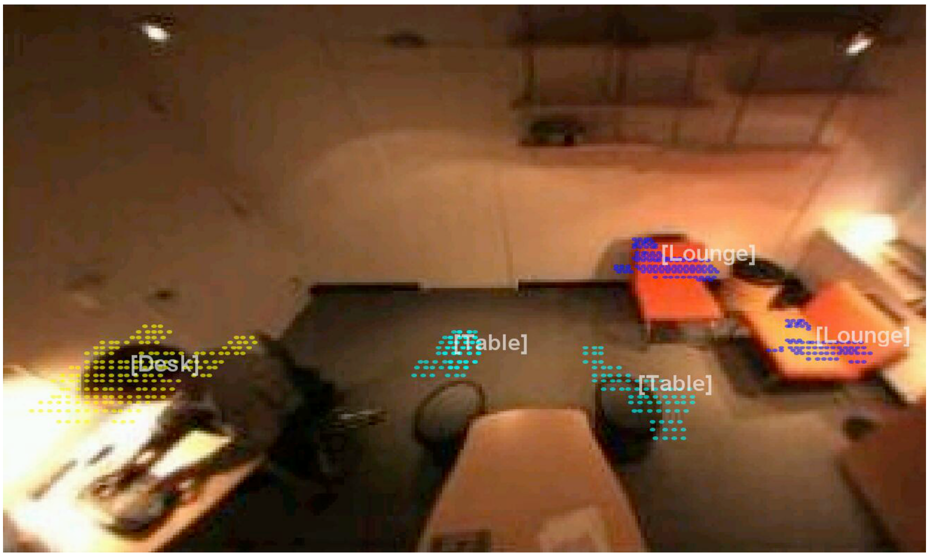

Fig. 9. The workspace from the camera's perspective, overlaid with clustered tracker data for three zones; labels (desk, table, lounge) are user-supplied

Fig. 9 shows the 2D extent for each of three activity zones clustered around furniture groupings. The tracker found a total of 11 zones. Using the zone editor, the user aggregated smaller zones into three zones of interest and pruned others. The table zone, for example, is an aggregation of zones around each of the stools. An access zone through the middle of the room was pruned. Fig. 10 shows an alternate zone map.

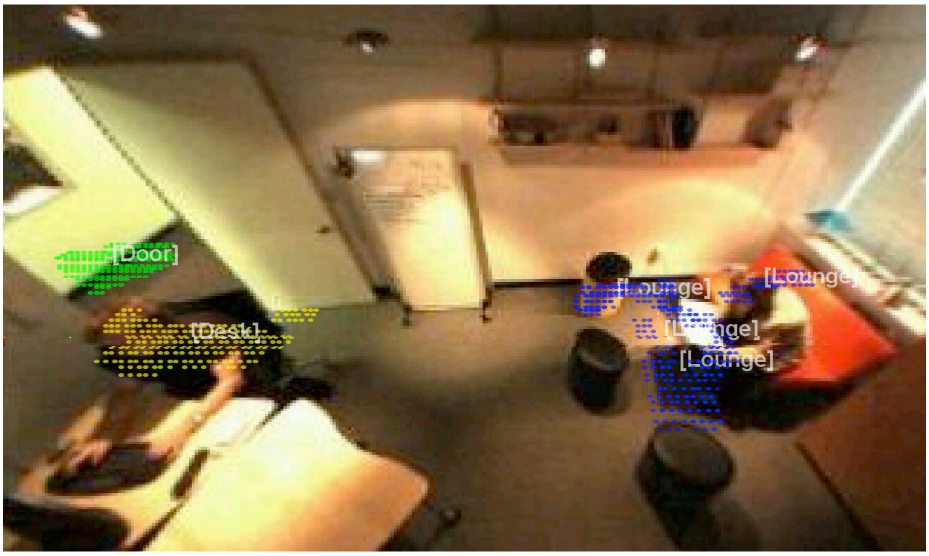

Fig. 10. The workspace overlaid with clustered tracker data for three different zones; zone labels (door, desk, lounge) are user-supplied

### 5.2 Blackboard and Infrastructure

Our perceive-reason-act paradigm is centered around a blackboard, which provides a shared memory through which system components communicate. (See Fig. 11.) The blackboard contains a context memory, a set of zone maps, a current zone map, a perceptual event queue, and a requested action queue. We use an implementation similar to that described in [26] and [33].[1] Perceptual systems, such as the tracker, post events to the blackboard. An agent-based system [9,18] provides the communication layer between the perceptual systems and the blackboard. By means of a publish-and-subscribe mechanism, the blackboard registers interest in particular classes of notifications, e.g., tracker events. Incoming events are added to the event queue. Once events are posted to the blackboard, a context inference system then reasons forward from the events, using such rules as "if there is more than 1 person in a zone then there is a meeting in the zone". Resulting inferences, such as "person 1 in a meeting", are posted to the blackboard as assertions in the context memory. In the terminology of a traditional blackboard model, the context inference system is represented as a set of knowledge sources, each of which is a set of rules contributing inferences at higher levels of abstraction than "raw" tracker events. These higher levels of abstraction allow users to specify preferences more naturally, e.g., in terms of a meeting rather than in terms of the number of people in a zone. They also enable preferences to work in the presence of dynamic activity zones since users can associate preferences with types of zones, e.g., meeting zones, rather than particular zones anchored in physical space.

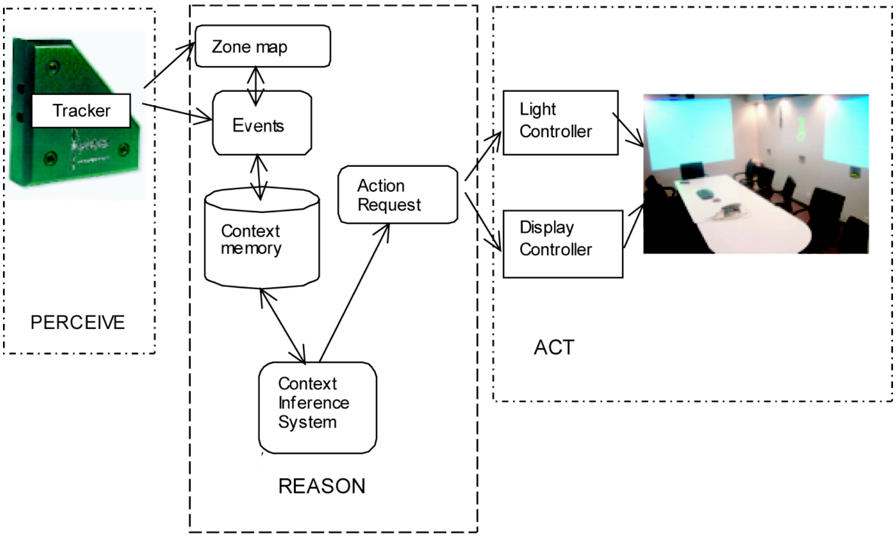

**Fig. 11. System architecture**

---

[1] Event queues can be inefficient because they force sequential processing of events. We have not found this to be a problem because most of our events thus far have come from a single stream of tracker data. See [26] for discussion of a principled approach to avoiding slow-downs due to event queues.

In addition to posting context assertions, the context inference system may post requests for action on an action request queue. In our current implementation, requests are processed sequentially by a central controller that aggregates similar requests and resolves conflicts between dissimilar requests. It, for example, will ignore a “turn off” request that is immediately followed by a “turn on” request for the same light. We currently are investigating the tradeoffs between centralized and decentralized control of device and application behavior.

### 5.3 User Interaction

As previously mentioned, the system is semi-automatic. It automatically determines zones and carries out preferred device and application behaviors. It relies on the user to: select and label zones of interest, add device and application behavior preferences to the zones, and define high level contexts, e.g., what constitutes a meeting. User interaction is via a simple graphical user interface, which we call the zone editor, that uses the displays shown in Figs. 9 and 10. Preferred behaviors are added to zones using a form template and a simple rule-like language. Examples of the language are shown in Fig. 12. Examples of specifying high level contexts, such as meetings, are shown in the next section.

```
if enter zone 0
then turn-on desk-lamp

if enter zone lounge
then turn-on lounge-lamp
     and turn-off desk-lamp
```

**Fig. 12. Examples of specifying preferred device behavior**

## 6 Example Applications

We describe two implemented examples of using activity zones as context cues. The first example, control of lights and audio, uses a direct mapping from events to environment actions. The second example, delivering a message based on context, demonstrates the use of inference to aggregate zones and to hypothesize high level context descriptions.

### 6.1 Device control

The system adjusts light and audio levels when people enter and exit particular activity zones. We used our tracking system to generate activity zones as described in the previous section. When we examined the resulting activity zone map, we discovered a small zone around each of the stools. We found it more useful to think of the stools as a single zone, especially when it came to specifying preferred device behavior. We aggregated the zones into a table zone using the zone editor mentioned in Sects. 5.1 and 5.3.

Typical device settings were:

- When someone enters the desk zone, turn on the lamp and music
- When someone enters the lounge zone, turn on the lamp in that zone and turn off the lamp in the desk zone
- When a second person enters the table zone, turn on the projector and turn off the music

When a person moved around in the room, tracker events for entry to and exit from zones were translated into requests for device control by means of the preference settings. Fig. 13 shows the state of lights and projector with two people sitting in the table zone.

We informally observed people working at the desk and meeting with colleagues at the table. We noted that the activity zones correlated well with particular activities, e.g., typing at the keyboard in the desk zone, reading and talking with others in the lounge zone, working alone or with others in the table zone. People described the zones by either the objects in the zone or by the activities that took place in the zones. Most would have preferred that zones be labeled automatically rather than by hand. Object recognition and activity inference could be used to provide such labels and to insure that labels carry semantic information rather than being just symbols. Informally, we found that people working in the room liked the organization of the room and thought the automatic control of lights, radio, and projector novel and useful. (A forthcoming memo will describe studies currently underway.)

**Fig. 13. Two people in the table zone. Lamp near desk shown at right has turned off; projector has turned on**

### 6.2 Message delivery

We implemented a message delivery system for context-aware notification in order to test inference of high level contexts, such as having a meeting, and for specification of delivery preferences. Examples of context inference and delivery preference rules are shown in Fig. 14. We also used inference rules for aggregation of several zones into a single larger zone as an alternative to hand editing of zones. Aggregation using rules works as follows. If we assume, for example, that we care about a zone C (e.g., near a table), that is the union of zones A and B (e.g., around two stools), then the inference system represents the aggregation of A and B using a rule such as “if a person enters zone A or zone B then the person enters zone C”. In essence, we use the rules to create a conceptual zone map that aggregates smaller zones into larger ones. When the tracker posts an event to the blackboard that a person has entered zone A, the inference system sees that event and infers that the person has entered zone C. If there are preferred application behaviors attached to entry in zone C, the inference system then can post the appropriate requests for action.

In our implemented example, we used rules such as those shown in Fig. 14. Rules 1 and 2 aggregate two stool zones into a table zone. Rules 3 and 4 specify that a person is in a meeting when the zone she is in contains at least one other person. Rules 5 and 6 specify message delivery preferences.

```
1. if person ?p in zone stool-1
   then person ?p in zone table

2. if person ?p in zone stool-2
   then person ?p in zone table

3. if number of people in zone ?z > 1
   then meeting in zone ?z

4. if person ?p in zone ?z
      and meeting in zone ?z
   then person ?p in meeting

5. if person ?p in meeting
      and message-delivery-event for ?p ?msg
   then deliver-without-display ?p ?msg

6. if deliver-without-display ?p ?msg
      and person ?p in zone ?z
   then notify agent for default-display-device for ?p in ?z
              "deliver-without-display" ?msg
```

**Fig. 14. Example context inference and message delivery rules (in pseudo-code); rules are implemented in Joshua [31]; ?x indicates a variable**

We had two people sit on the stools, and we sent a message to one of them, which caused a message delivery event to be posted to the blackboard.[2] The tracker noticed a person in each of the two stool zones; the context inference system inferred that each person was in the table zone and that there was a meeting in that zone. The message then was delivered without being displayed—it appeared as an icon on the computer screen in the desk zone, which was the default display device for the recipient when she was in a meeting. With one person in the room, a message is displayed on the computer screen if the person is in the desk zone (where the screen

---

[2] People were identified by the system with an explicit utterance.

is), or projected on the wall if the person is in the lounge zone. Anecdotally, experiences of novice users (computer science graduate students) suggest that the system is an effective way to adjust notification state. Users need not make explicit gestures or utterances to change notification state; they need only specify the preferred behavior of the environment. (A forthcoming memo will describe studies currently underway.)

The above description illustrates two important points: (1) activity zones can be used to deliver context-dependent information, and (2) the blackboard and context inference system enable user-specified aggregations of zones and delivery preferences stated in terms of high level context descriptions.

## 7 Discussion and Future Work

The concept of activity zones—regions of location context formed by observing user behavior—is a broad one, and we have only begun to explore its full extent. In addition to ongoing user studies of the current prototype, we are continuing to explore issues of user interaction focus on such questions as: which device and application behaviors are useful, how do users specify those behaviors, how zones are displayed to users. We also are exploring the issues of how our activity zone system selects default preferences for anonymous people, or selects preferences in a space inhabited by multiple people, each of whom may have their own preferences.

We anticipate being able to use information about furniture identity and locations to augment our context inference system with simple activity models representing such information as "meetings often happen at tables", "informal meetings often happen in comfortable chairs". We could use the perceptual features from our tracking system, plus identity of furniture objects, to index into a catalog of higher level contexts representing such activities as being in a meeting or reading (e.g., as in [32]). With extra information about objects and activities, we would be able to infer users' activities more accurately, thus increasing the relevance of task-related information and services provided by the workspace.

In building a catalog of higher level contexts, we plan to augment our observations of people in everyday work situations by investigating activity theory [5, 16], research on understanding how people work [14, 29], and how workspace design affects people's work [14, 20, 34]. This body of literature, along with our observations, will provide us with examples of social interactions that help define context for people.

An activity zone map may not be relevant if a physical space is rearranged, as shown, for example, in Figs. 9 and 10. In such circumstances, the system must adapt the activity zone map. One could consider two approaches: either save an image of a space for each of several activity zone maps, and periodically check that the image matches the current furniture configuration; or record furniture type and location in or near particular zones, then use a furniture tracker to notice a mismatch between original furniture locations and new locations. Since furniture tracking is a challenging computer vision problem, the first approach may be easier to implement. With easily distinguishable furniture, the second approach may be advantageous since it allows for more flexibility in space layout. We plan to explore both of these approaches.

We are keenly aware that many people are uncomfortable having their activities observed by cameras, and privacy issues deserve attention when designing context-aware systems, e.g., [1]. In our work to date we have found that it is important to have computationally transparent perceptual systems so that users are aware of the kind of image information that is processed and stored. Abstract trajectory data with voluntary and explicit identity assertion is generally considered less invasive of privacy than full motion video. We plan to investigate methods for real-time visualization of tracker state.

We have shown that our concept of activity zones can provide valuable context cues for context-aware applications. So far we have explored this idea with relatively simple applications—device control and message delivery. We plan to continue our investigations by using activity zones with other applications, for example, retrieving information based on a predicted context. A zone's information about motion could be used to predict where people are likely to be sitting or standing. This predictive capability may prove valuable with tasks that are more structured than those usually found in an office environment. In a laboratory setting, for example, scientists often engage in an ordered set of activities, centered around various pieces of laboratory equipment. Context-aware computing researchers are building applications to support such laboratory activities, e.g., [4]. The activity zones in such a setting would center on laboratory equipment stations, and could be mapped to an ordered list of activities in order to predict likely transitions between zones. By anticipating what scientists may do next, a support system could initiate start up procedures when necessary or gather additional information needed at a particular station.

Similarly, we believe that the activity zone concept has utility in the structured domain of theatre performance. We are in the beginning stages of a collaboration with the Royal Shakespeare Company, and plan to explore the use of the activity zone concept in tracking actors, identifying particular scenes in the progression of a play, and controlling lighting [28]. We also are talking with physicians at a local hospital about the possibility of using the activity zone concept to track the progression of activities in an operating room.

In these and other applications, we believe that activity-based location regions provide a richer level of context information than has been previously available in location-based context-aware computing.

## Acknowledgments

We thank members of the AIRE (Agent-Based Intelligent Reactive Environments) and Visual Interface groups for help in setting up the workspace and for their insightful conversations about this research. We also thank Randall Davis and Robert Laddaga for valuable comments on drafts of this paper. This research is sponsored by MIT Project Oxygen. We are grateful for support from the MIT Oxygen Alliance Partners: Acer, Delta, Hewlett-Packard, Nokia, NTT, and Philips.

# References

1. Ackerman, M., Darrell, T., and Weitzner, D.J.: Privacy in Context. Human-Computer Interaction 16 (2001) 167-176.
2. Addlesee, M., Curwen, R., Hodges, S., Newman, J., Steggles, P., Ward, A., and Hopper, A.: Implementing a Sentient Computing System. IEEE Computer 34 (2001) 50-56.
3. Alexander, C., Ishikawa, S., and Silverstein, M.: A Pattern Language: Towns, Buildings, and Construction. New York: Oxford University Press (1977).
4. Arnstein, L.F., Sigurdsson, S., and Franza, R.: Ubiquitous computing in the biology laboratory. Journal of the Association for Laboratory Automation (JALA) 6 (2001).
5. Bannon, L.J. and Bødker, S.: Beyond the Interface: Encountering Artifacts in Use. In: Designing Interaction: Psychology at the Human-Computer Interface. J.M. Carroll (ed.). Cambridge University Press, New York (1991) 227-253.
6. Beigl, M. (ed. Special Issue on Location Modeling in Ubiquitous Computing. Personal Ubiquitous Computing. Vol. 6 (2002).
7. Brumitt, B., Krumm, J., Meyers, B.B., and Shafer, S.: Ubiquitous Computing and the Role of Geometry. IEEE Personal Communications 7 (2000) 41–43.
8. Brumitt, B., Meyers, B., Krumm, J., Kern, A., and Shafer, S.: EasyLiving: Technologies for Intelligent Environments. In: Proceedings of the 2nd International Conference on Handheld and Ubiquitous Computing: Springer-Verlag (2000).
9. Coen, M., Phillips, B., Warshawsky, N., Weisman, L., Peters, S., and Finin, P.: Meeting the Computational Needs of Intelligent Environments: The Metaglue Environment. In: Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments (MANSE 99). Dublin, Ireland: Springer-Verlag (December 1999).
10. Crabtree, A. and Rodden, T.: Routine Activities and Design for the Domestic. University of Nottingham Equator-02-037 (2002).
11. Demirdjian, D., Tollmar, K., Koile, K., Checka, N., and Darrell, T.: Activity Maps for Location-Aware Computing. In: Proceedings of IEEE Workshop on Applications of Computer Vision (WACD2002) (December 2002).
12. Dey, A.K. and Abowd, G.D.: Towards a Better Understanding of Context and Context-Awareness. In: Proceedings of the What, Who, Where, When, and How of Context-Awareness Workshop,CHI 2000: ACM (2000).
13. Dey, A.K., Abowd, G.D., and Salber, D.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction 16 (2001) 97-166.
14. Duffy, F. and Powell, K.: The New Office. London: Conran Octopus Ltd (1997).
15. Engelmore, R. and Morgan, T. (eds): Blackboard Systems. Addison-Wesley, Reading, M A (1988).
16. Engestrom, Y., Miettinen, R., and Punamaki, R.-L. (eds): Perspectives on Activity Theory. Cambridge University Press, New York (1998).
17. Hall, E.T.: The Hidden Dimension. New York: Doubleday (1969).
18. Hanssens, N., Kulkarni, A., Tuchida, R., and Horton, T.: Building Agent-Based Intelligent Workspaces. In: Proceedings of The International Workshop on Agents for Business Automation. Las Vegas, NV (June 2002).
19. Harle, R.K. and Hopper, A.: Using Personnel Movements for Indoor Autonomous Environment Discovery. In: First IEEE International Conference on Pervasive Computing and Communications (PERCOM 2003). Fort Worth, TX: IEEE (2003).
20. Hartkopf, V., Loftness, V., Aziz, A., Shankavaram, J., and Lee, S.: The GSA Adaptable Workplace Laboratory. Lecture Notes in Computer Science 1670 (1999) 29-44.
21. Hertzberger, H.: Lessons for Students of Architecture. Rotterdam: Uitgeverij Publishers (1991).

22. Hightower, J. and Borriello, G.: A Survey and Taxonomy of Location Systems for Ubiquitous Computing. University of Washington, Computer Science and Engineering: Seattle, WA (2001).
23. Hightower, J., Brumitt, B., and Borriello, G.: The Location Stack: A Layered Model for Location in Ubiquitous Computing. In: Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002). Callicon, NY (2002).
24. Hildebrand, G.: The Wright Space: Pattern and Meaning in Frank Lloyd Wright's Houses. Seattle: University of Washington Press (1991).
25. Hillier, B. and Hanson, J.: The Social Logic of Space. New York: Cambridge University Press (1984).
26. Johanson, B., Fox, A., Hanrahan, P., and Winograd, T.: The Event Heap: A Coordination Infrastructure for Interactive Workspaces. In: Fourth IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2002). Callicoon, NY (2002).
27. Junestrand, S., Keijer, U., and Tollmar, K.: Design Patterns for Private and Public Digital Spaces in the Home Environment. International Journal of Human-Computer Studies 54 (2001) 753-778.
28. Kennan, F.: Lights, Hologram, Action! Business Week, May 26 (2003) 97.
29. Kirsh, D.: The Context of Work. Human-Computer Interaction 16 (2001) 305-322.
30. Lahlou, S.: Observing Cognitive Work in Offices. In: Cooperative Buildings: Integrating Information, Organizations, and Architecture. N. Streitz, et al. Editors. Springer, Heidelberg (1999) 150-163.
31. Rowley, S., Shrobe, H.E., Cassels, R., and Hamscher, W.: Joshua: Uniform Access to Heterogeneous Knowledge Structures, or why Joshing Is Better than Conniving or Planning. In: AAAI (1987).
32. Turner, R.M.: Context-Mediated Behavior for Intelligent Agents. Int. J. Human-Computer Studies 48 (1998) 307-330.
33. Winograd, T.: Architectures for Context. Human-Computer Interaction 16 (2001) 401-419.
34. Zelinsky, M.: New Workplaces for New Workstyles. New York: McGraw-Hill (1998).