Journal of Universal Computer Science, vol. 14, no. 17 (2008), 2840-2858
submitted: 30/6/08, accepted: 27/9/08, appeared: 28/9/08

Authoring Social-aware Tasks on Active Spaces

Roberto F. Arroyo
(Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada, Spain
robfram@ugr.es)

Miguel Gea
(Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada, Spain
mgea@ugr.es)

José Luis Garrido
(Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada, Spain
jgarrido@ugr.es)

Pablo A. Haya
(Departamento de Ingeniería Informática
Universidad Autónoma de Madrid, Spain
Pablo.Haya@uam.es)

Rosa M. Carro
(Departamento de Ingeniería Informática
Universidad Autónoma de Madrid, Spain
Rosa.Carro@uam.es)

Abstract: Social-aware computing is an emerging trend based on ubiquitous computing technologies and collaborative work. A successful design demands a better understanding of group tasks, adaptation mechanisms and support for dynamic changes in a nomadic computing paradigm. This paper proposes the use of a hypermedia model to describe and support group activities in intelligent environments. The resulting system integrates adaptive context-aware information on the basis of user/group models in order to provide a structured access to dynamic task scheduling. In particular, we propose the use of the calendar metaphor as an ongoing connection between active spaces and collaborative tasks. This proposal provides the appropriate support for an easier human coordination to achieve common objectives in blended learning scenarios, and thus, extending authoring social tasks to physical spaces.

Key Words: Hypermedia Authoring, Collaboration, Ambient Intelligence, Active Spaces, Task Scheduling, Social-aware computing
Categories: H.5.1, H.5.2, H.5.3, H.5.4, K.3

1 Introduction

Social-aware computing is an emerging trend based on understanding "how other people's activities are part of the context of one's own activities" [Dourish and

Bellotti 92]. It has been successfully followed in different types of Web-based community models (on-line communities, wikis, blogs, instant messaging, or business models), supporting an effective human collaboration [Gutwin and Greenberg 04]. Connections with the physical world represent a further step, facilitating the understanding of different activities, as well as spatial-temporal relationships between them. In order to improve user's experience it is necessary to pay special attention to the integration of information interfaces into the actual, ubiquitous environments; many invisible computers and devices are available throughout the physical environment allowing access to shared information at any time and place [Weiser 91]. This approach is rooted on the Ambient Intelligence [Ducatel et al. 01] paradigm. It constitutes an advance in the user-centred approach for computer applications, adding natural interaction on computerised environments, also called Active Spaces.

Within communities, explicit collaboration is indispensable to achieve common goals. Context information is required for a better understanding of other users' activities, locations and identities, along with current and future plans, and resource status (available resources). Besides, users should access the information the way they wish. Therefore, it seems reasonable to use Adaptive Hypermedia [Carro 99] taking into consideration context information (devices and locations) and user's preferences.

The work presented in this paper focuses on supporting communities in nomadic computing environments taking into account contextual information, mainly especially related to actors, activities and resource management. It makes possible the creation of a system supporting and managing task scheduling in collaborative intelligent environments [Martin et al. 06]. For such purpose, task models [Diaper 02] have been created to represent user's activities and scheduling, as well as resources being shared by various users [Garrido et al. 05]. A suitable representation of this domain is required to identify each relevant and meaningful entity involved in context adaptation. In that sense, AH allows tailoring the domain content based on user's model and context.

This proposal will be applied to a blended learning (b-learning) case study in which on-site traditional classes and on-line activities are performed. In such as a b-learning scenario the following circumstances are present [Abowd 99]: i) users play different roles (teacher, student or staff), ii) several kinds of tasks are carried out (meetings, classes, seminars...), iii) tangible resources must be handled (classes, electronic whiteboards, laptops, PDAs, projectors), and iv) different media are available (slides, hypermedia pages, exam-copies, etc.). The use of models representing all these elements is necessary for managing context awareness and task resources, but also mechanisms for hiding irrelevant information to users are needed. In order to address the complexity of such scenario, we propose the use of a calendar metaphor based on hypermedia navigation to aid users to schedule contextualised tasks in Ambient Intelligence environments. In particular, the proposal addresses the representation and management of groups (roles, actors and policies), activities, and other key context-aware information (such as location, identity, and resource state) to support and assist current and future learning activities. This management allow us to consider social awareness in physical spaces as a new dimension to design learning tasks using adaptive hypermedia models.

The rest of the paper is structured as follows. Section 2 presents the basis of this research focused on collaborative task modelling and development of Ambient

Intelligence systems, in which the calendar metaphor is used for a context-aware activity scheduling. Section 3 shows the integration scheme of this research in a layered architecture using hypermedia as the underlying model to preserve social-awareness information. Section 4 describes the implementation of this proposal in a laboratory at the Universidad Autónoma de Madrid. Finally, Section 5 summarises the main contributions and outlines future research.

2 Background

This section introduces the three cornerstones of this research work in the following order: social scheduling techniques using the calendar metaphor, task-based modelling, and Ambient Intelligence (AmI) systems.

2.1 Scheduling Using Calendar Metaphor

Social-awareness implies understanding the context of a group when its members work together [Erikson and Kellogg 00]. Difficulties arise when groups belong to wide and complex organisations. In this context, users share common tasks playing different roles with specific time/space matrix constrains (e.g. synchronous and long-life collaboration in distributed spaces). On the one hand, time represents a granular approximation to tasks (past, present and future), giving information about which users and resources are involved in each activity too. On the other hand, space dimension suggests the need of a location mechanism, for example, to find out resource availability. Hence, the level of detail of the information to be managed constitutes another issue to take into consideration.

Calendar metaphor is a well-known simile that has traditionally been perceived as both individual and group tool for task scheduling (events, meetings, and so forth) using different granular temporal dimensions for planning [Beard et al. 90]. The digital family calendar [Mackinlay et al. 94] is a physical digital resource providing good results as a planning tool. Visualization [Bardram and Hansen 04], connection with physical world [Neustaedter and Bernheim Brush 06] and social awareness (knowing the other's related task) are important challenges to be addressed. In this direction, for example, calendars that include a basic location-aware mechanism for family members usually provide an adequate group-aware coordination support.

Simple changes in real world, such as somebody enters/exits a room and new events to be inserted, may affect the plans already registered in the calendar or current or future activities. Therefore, the tight interconnection between the calendar (task world) and physical world requires the support of a dynamic model allowing users to be aware of related tasks. Moreover, a calendar contains a temporal task representation of each entity (user, artefact) with information about the physical location for activity accomplishment.

In b-learning scenarios, activities are represented as tasks in the calendar. Different actors share common tasks (lectures, tutoring sessions, practical tasks, etc.) to be accomplished in active spaces. The system is aware of context (i.e. current user location). When dealing with variable information, it is necessary to manage change propagation properly and consistently by adding, modifying or removing relationships between entities, due to task are context dependent. For example, in our sample b-learning scenario, Mairi (the teacher) decides that next week, an extra face-to-face Programming course class will be held in a laboratory (instead of in the usual classroom), in order to solve practical problems. Decision support is a basic social aware mechanism to help her in this situation in the following way: Which are the free time slots for Mairi? And which are those for the students? Which laboratories are free at that time? Do these laboratories have the required characteristics and resources?

Social awareness requires the understanding of common tasks and user's context to manage these situations dynamically. As resources are shared by different users, once solution to that dynamic new situation has been found, for example, when a free time slot and the suitable laboratory to hold the class have been found,, all the users should be able to observe the resulting state, that is, teacher and students should be notified about the assigned date and class. Scheduling and planning bearing in mind preferences or restrictions have been widely studied [Brzozowski et al. 06] through a range of different techniques. Scheduling refers to allocation and cancellation tasks. Since previous scheduled tasks (i.e. to give classes) may change according to other new scenarios, it is important to implement an efficient notification mechanism and know the general context in order to understand current states and future availabilities.

Context awareness means that changes in current active spaces can be detected, identified, represented and reflected. The inherent complexity managing context awareness is due to various reasons. Firstly social communities usually apply rules and common practices to solve problems on the basis of, for instance, priority policies for fulfilling specific requirements (e.g., taking into consideration teacher's restrictions before students' ones). Secondly,usability implies that information should be easy to use and understand, and should also be adapted to user's preferences and devices to be used (PDA, phone, etc.). Finally, time should be managed at different levels to support better the variability in task duration [Ning et al. 02]: temporal planning can cover days, weeks or even semesters, and therefore a proper level of detail is needed in each case (hiding irrelevant information). The fuzzy nature of human time [Payne 93] must also be considered. In [Kutar et al 00] is addressed the difficulty of managing statements with both contextual and fuzzy semantics, such as the precise meaning of the statement "on the same day".

When people use calendars to schedule tasks or other types of appointments, they unconsciously perform different operations on the calendars. For example, when two people are trying to arrange a meeting in a given week, they try to find a common free slot in both calendars. Unconsciously, what they do is to combine both calendars for that week, and search for free slots. Similarly, we can outline two additional operations [Harris and Stocker 98]: intersection and difference. The former is useful to see common events between calendars, and the latter is used to exclude certain kinds of events from a calendar.

The tasks are common to more than one personal calendar, which is important for a correct understanding of calendar operation within the system. The union operation denotes the set of task that are obtained from the combination of two or more calendars (each cell may contain one or more activities denoting parallel activities). Intersection obtains common tasks for the same periods of time. Difference represents the set of activities that does not overlap with other planned activities, that is, the set of activities that could be shared in both calendars if it is necessary.

2.2 Task-based Model for Community Support

Systems supporting complex social interactions and dynamics require specific models to represent collaboration and behaviour. In order to address their complexity, we adopt the task- and behaviour-based approach provided by the AMENITIES [Garrido et al. 05], a conceptual and methodological framework for modelling cooperative systems.

Figure 1(a) shows the concepts present in the conceptual framework AMENITIES, as well as their relationships, using a UML class diagram. In this conceptual framework, an action is an atomic work unit. Its event-driven execution may require, modify or generate explicit information. A subactivity is a set of related subactivities and/or actions. A task is a set of subactivities intended to achieve a certain goal. A role is a descriptor for a set of related tasks to be carried out. An actor is a user, program, or entity with certain acquired capabilities (skills, category, etc.) that can play a role in the execution of actions (using artefacts). A group performs certain subactivities depending on interaction protocols. A cooperative task must be carried out by more than one actor. A group is a set of actors playing roles that perform one or more cooperative tasks. A group may comprise subgroups. A law is a limitation or constraint imposed by the system which is used for adjusting the set of possible behaviours dynamically. An event represents the occurrence of a significant fact in the system. An organization consists of a set of related roles. Finally, a cooperative system is composed of organizations, groups, laws, events and artefacts.

In [Arroyo et al. 06], the conceptual model for a specific system is translated into a structured semantic network representing concepts (entities) and relationships. Figure 1(c) shows the semantic network generated with this model for an example of class arrangement, where elements from new and conventional models (e.g. spatial location and task models respectively) are considered. The figure shows some elements related to the actor Mairi, who is playing the role *Teacher* while carrying out the *GiveClass* task.

2.3 Active Spaces

The goals of this research is the development of active spaces for social aware task scheduling, which was a key topic to be addressed in project U-CAT (Ubiquitous Collaborative Adaptive Training) [UCAT]. In this project, a framework was developed to support a seamless integration among different pervasive components from different heterogeneous technologies (physical devices and TCP/IP networks) using adapted interaction and natural interfaces [Montoro et al. 04]. As ubiquitous systems, Ambient Intelligent environments are composed by an infrastructure of sensors and actors. With the aim of separating low-level sensor data processing from high-level applications, we need to introduce a middleware layer that fetches data from sensors and converts them into a format comprehensible before distributing to other related applications. Using this middleware, labs are controlled through Web interfaces or natural spoken language [Montoro et al. 04b], as illustrated in Figure 1.b.

The proposed system supports a context layer based on the blackboard metaphor [Haya et al. 04], which stores a world model using a semantic network as described in next section. This model stores all the relevant information and is used for asynchronous information querying. Figure 1.c) shows the semantic model and the mapping between entities and physical resources. Context-aware information is maintained in this model since changes are automatically reflected on it. Therefore, each time the blackboard is accessed, the semantic network is queried.

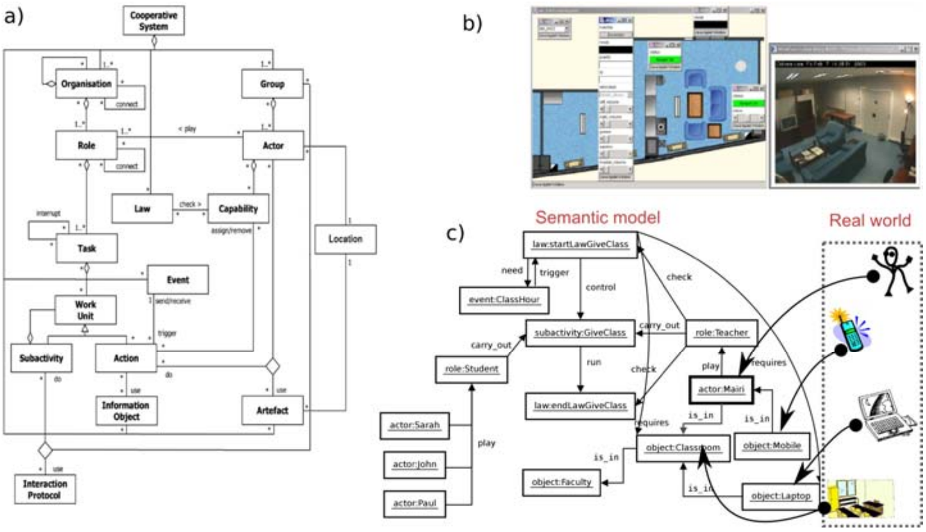

Figure 1: a) AMENITIES, main concepts for task modelling; b) The U-CAT laboratory: Web interface and on-line camera screen; c) Semantic model entities corresponding to physical entities

Regarding ubiquitous systems, some quality attributes that characterize them are: scalability, since a b-learning system can contain more than a hundred participants and the system must manage only relevant information [Mackinlay et al. 94] with an expected time response which should be independent of the number of users; privacy [Boyle 03], since information should follow a restrictive policy, and must also administer time intervals for private use (users can decide whether to make tasks visible to other ones); and remote notification, since information may be obtained anywhere at any time. Notification mechanisms can be delivered according to different strategies and criteria [Andueza and Carro 06] to represent changes in task scheduling according to personal preferences (e.g. user's status) or system congruence (e.g. change of labs state) to prevent annoying situations [Werle and Jansson 02].

3 A Proposal for Interfacing Social-Awareness

This section describes a smart system that uses adaptive hypermedia to support navigation through a semantic model connected with the physical world. Its benefits are user friendliness, adaptation capabilities and, especially, that it constitutes a valuable help for group awareness activity design, since it can be used as support for authoring activities in the context of active spaces.

The system consists of three layers, as shown in Figure 2. The first layer, called *the calendar layer*, represents the user interface and offers scheduling functionality to the community. This calendar model relies on the second layer, which is based on *the hypermedia network*. The hypermedia network retrieves and organizes the information contained in the third layer, *the semantic model layer*, which is in charge of supporting the navigation mechanism. In this layer, the underlying information about physical active spaces is also associated with each task and its other related concepts (roles, actors, etc.).

The calendar metaphor provides a view of the system (users, activities and resources) using a typical time slice. The user can navigate through hypermedia whilst perceiving temporal and semantic relationships. Each entry is linked to a hypermedia node (e.g. ClassHour event.) For example, Mairi gives a lecture in classroom A2 at 12:00 on Monday. She annotates the meeting in the system. Relationships are, therefore, created between Mairi and other entities: the teacher role, the classroom A2 and Monday at 12:00.

The system behaviour is addressed in two different ways. Firstly, entities are physically located on active spaces, so changes on, for instance, location, are directly reflected in the semantic network. Secondly, actors (users or agents) schedule future tasks using the calendar, tagging new information directly on the hypermedia model, which is led to the semantic network as future events to be also scheduled.

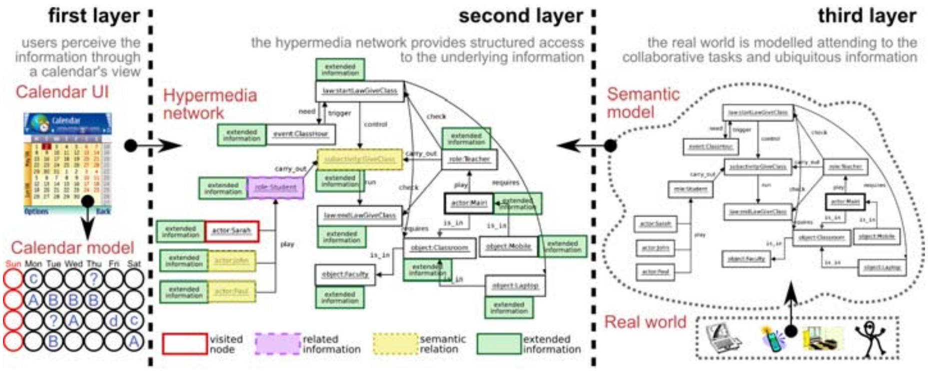

Figure 2: The system is structured in three layers: calendar-based user interface, hypermedia network and semantic model layer

The composition of these layers and the transitions between them, as well as the semantic network filtering through the calendar, are explained in the following subsections.

3.1 From Calendar to Hypermedia Network

The calendar-based interface offers functionality for social scheduling and filtering. The calendar gathers relevant information about users, tasks and resources using pointers to the corresponding hypermedia nodes. The hypermedia nodes link to their corresponding semantic elements in the underlying layer. In this way, dynamic modifications in the semantic storage are instantly reflected into the hypermedia network and also in the calendar layer. Thus, the calendar presents a time-based view of the hypermedia network. Calendar cells are filled with pointers to the corresponding elements (tasks, roles, actors, objects...), and the user chooses the desired time slices or elements in order to proceed.

Task scheduling starts from users' actions on their calendars. For example, when two people are trying to arrange a meeting at a given week, they try to find a common free slot in both calendars. This implies to match both calendars in order to obtain free slots for that week, which is supported by the system. Two additional operations can also be used to obtain different information [Arroyo et al. 06]: intersection (to find common events between calendars) and difference (to exclude certain kinds of events from a calendar).

Filtering is essential to navigation. Thus, the user can select the information to be showed. Groups of interest and node restrictions can be established. The calendar supports filtering by taking into account different types of restrictions such as primary elements (actors, tasks, locations...), relations (actors playing specific roles, tasks requiring certain objects...), dates (next week, tomorrow, less than a month, next hour...), etc.

The calendar interface must display readable information, and the hypermedia network provides it. In order to support navigability, information related to neighbour elements must also be retrieved. This process is done in three stages, as represented in Figure 3. Firstly, direct information is retrieved (i.e. for a given actor, his/her icon or surname). Secondly, the related information is obtained from the semantic relationships, for example, for a task the laws to be satisfied before starting. And finally, the information is retrieved navigating through relevant semantic links (i.e., in order to get the tasks in which an actor is taking part, one needs to navigate from actor to role and from role to tasks).

Additionally, a navigation-awareness mechanism has been incorporated to let the system itself be conscious about who is navigating at each time and which node is being visited. This hypermedia network stems from applying a simple mapping between model entities and hypermedia elements, generating a view-like explicit hypermedia network through which the users can navigate.

The semantic network is in charge of providing adaptation to the user. Different actors may have different preferences about displaying the same information. Moreover, each different role that a certain user can play has additional preferences associated. It is possible, then, to provide adaptation to different users that play the same role. In addition, every type of task may be navigated in a different manner attending to its different nature. Therefore, preferences for tasks are also stored. The most appropriate set of information and navigation links for each user, according to the role and task he is visiting, can be presented at each time.

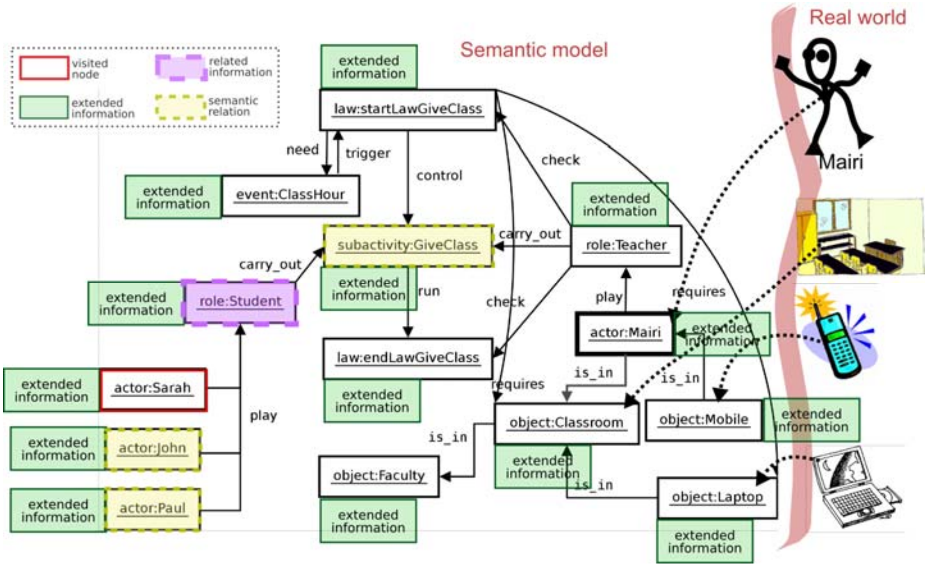

Figure 3: The hypermedia network is generated by adding additional information to the semantic network

Each new entry in the calendar is mapped onto a task pattern in the hypermedia model. For example, task "GivePresentialClass" automatically defines an entry point connected to the corresponding nodes (where:location, when:date, duration:time, members:students, material:slides). This information can be filled in with context-aware information, i.e., a free class chosen automatically for that date (querying available ones), or a common free hour selected between those available at each student's calendar. Thus, when Mairi annotated the class in classroom A2 on Monday at 12:00 a new entry was inserted in her calendar. Additionally, entries in the classroom and students calendars were created too. All the entries have their corresponding relations with other entities. Mairi accesses to this information through a calendar's view, showing the classroom icon, student photographs and additional textual information obtained from the hypermedia network: actors, roles, tasks and other objects are connected to a related hypermedia node with this kind of information.

3.2 From the semantic network to the hypermedia model

The hypermedia network is built on top of the semantic network defined by the design model. Semantic network nodes are mapped to hypermedia nodes, and the relationships between them are mapped to hypermedia links that can be used for navigating. Using this consideration, one can think on an initial hypermedia network in which nodes represent actors, tasks, roles, etc. Starting from a node, one can navigate towards semantically related nodes. Changes in context (e.g. a teacher is entering the classroom) produce changes in the entity location, which are consequently propagated to the hypermedia network.

Interesting feedback is provided when the semantic network is updated as a result of changes. For example, when the class was planned and inserted in the calendar, 12:00 was set as the starting time. In fact, the “Class time” event reminds Mairi and her students the class starting. Yet the class starts when Mairi “enters” classroom A2 (Mairi's location is updated). At this precise time, the hypermedia model represents the running task in Mairi and students' calendar, and each one knows the availability of the rest. This information is especially useful for other concurrent tasks, such as meetings or seminars. In such cases, even if the activity had been previously planned, if the location of, i.e., the department chair or the tutoring teacher is unexpectedly in class (or giving a seminar), his state will change automatically to busy and the others will be aware of this fact.

3.3 Unifying Semantic Network Filtering Through Calendar

A design model [Arroyo et al. 06] has been proposed on the basis of the AMENITIES conceptual and methodological framework [Garrido et al. 05]. This model includes tasks as first class elements for representing context information in AmI systems. In order to correctly integrate a scheduling system within a task model in AmI environments, a semantic model should be used. Necessary flexibility can be achieved by calendar tools, and even intrusive and proactively reduced, making use of the increased knowledge about the activities occurring in the system. Figure 4 shows an example of the resulting semantic network in which common elements (e.g. spatial location and task model elements) are considered, comprising first class elements for information context, as well as the desired link between calendar scheduling and the system model, and the correspondence between the latter and the real world. The figure also shows Mairi’s calendar, in which the planning of a future class has been highlighted. As this constitutes a relationship (not only a label) between the task and the time slot, a richer semantic model has been created to provide detailed information about actors involved in tasks, roles they are playing, required resources, etc.

We emphasize the benefits from semantic web in AmI environments for implicit scheduling, decreasing the overhead when users arrange tasks. Since our AmI model establishes a semantic network which combines elements from ubiquitous and task-based models, when a task is defined, this network can be explored by navigating through the links and extracting the relevant information. For example, if we plan a class in classroom 3-A on Wednesday with Mairi as a teacher, even though Mairi notes down that this classroom is booked, we can create a calendar for classroom 3-A by showing the people who will be in it, using the information provided by all the users to complete new calendars. We should not forget that the system is dynamic and that users may change their position frequently. User location can be traced using these calendars, since they store not only future events, but also past and present ones.

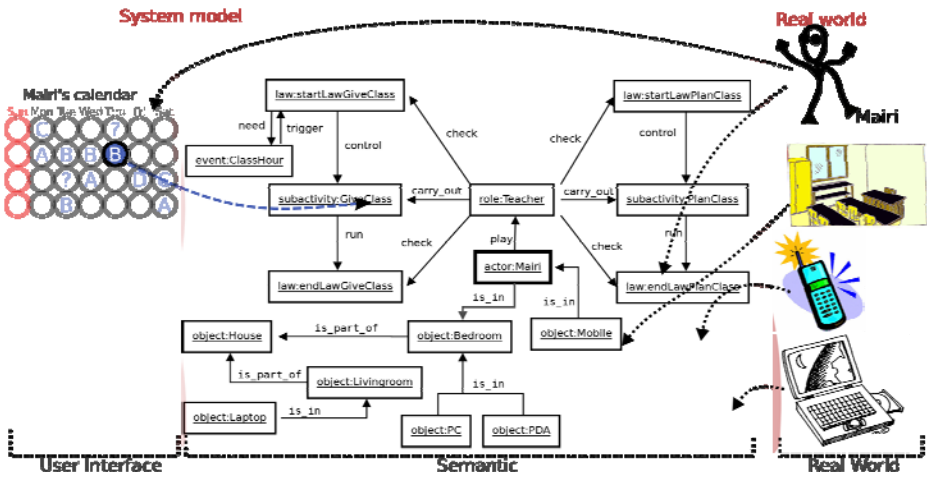

Figure 4: The calendar annotates tasks with semantic information, providing a familiar interface.

Once the conceptual model for contextual information has been presented, the way in which different calendars can be built from the semantic network is explained as follows. A calendar is useful not only for users, since the calendar definition includes other entities. We consider the entire semantic network to be a set *S* and any calendar *C* is a subset of *S*. We define the following two attributes for any calendar *C*:

- Source: Entity interested in the calendar. Sources are denoted with superscripts (C^{source}). For example, we can write C^{actors} to define those entities (actors) who will be interested in the calendar. Attribute-value pairs can also be assigned to the source list. For example, $C^{\text{actors}\rightarrow\{\text{name=Mairi}\}}$ defines a calendar in which Mairi is the selected entity. Relations *equal to* and *different from* can be used, as well as a list of values for the attribute. A specific source must be defined to create a calendar.
- Target: Entity that is important for the source. In other words, targets are entities aimed to be recorded in the calendar's time slots. Targets are denoted with subscript (C_{target}). If the target is not present, then all the entities are interesting for the source. For example, $C^{\text{actors}}{}_{\text{tasks}}$ will construct a calendar for actors in which their associated tasks comprise the entities to be noted down on it. Attribute-value pairs can also be specified, the same as in sources. For example, if someone wants to construct a calendar for actors containing any task except PlanClass, the calendar $C^{\text{actors}}{}_{\text{tasks}\rightarrow\{\text{name}\neq\text{ PlanClass}\}}$ must be defined.

Figure 5 represents the construction of the semantic network associated to calendar queries. We can establish the following equivalences between this notation and the previously used algebraic notation:

1. If a calendar contains Mairi's tasks (C_a or $C^{\text{actors}\rightarrow\{\text{name=Mairi}\}}{}_{\text{tasks}}$) and another different one contains Pádrig's tasks (C_b or $C^{\text{actors}\rightarrow\{\text{name=Pádrig}\}}$), we can define $C_a \cup C_b$ as $C^{\text{actors}\rightarrow\{\text{name=(Mairi,Pádrig)}}{}_{\text{tasks}\}}$.

2. If a calendar contains Mairi's tasks, the intersection of the calendar containing all actors' tasks (C_a or $C^{\text{actors}}{}_{\text{tasks}}$) and the calendar containing Mairi as an actor (C_b or $C^{\text{actors}\rightarrow\{\text{name=Mairi}\}}$) is defined as $C_a \cap C_b = C^{\text{actors}\rightarrow\{\text{name=Mairi}\}}{}_{\text{tasks}}$. Similarly, if someone wants to get information about tasks in which all the actors, except Mairi, are involved, the corresponding sentence to be built is $C_a \setminus C_b = C^{\text{actors}\rightarrow\{\text{name}\neq\text{Mairi}\}}{}_{\text{tasks}}$.

This definition presents a degree of freedom that allows the definition of multiple views of the system as different calendars, it covers different needs by relying on the same information in a homogeneous way. For example, if someone wants to know where Mairi will be, a calendar such as $C^{\text{actors}\rightarrow\{\text{name=Mairi}\}}{}_{\text{locations}}$ can be generated. This sentence will create a calendar where Mairi is the actor. In this case, the interest is put on the source, and the location entities (target).

It can be concluded that the defined calendar extends the functionality delivered by conventional ones. By separating entities and different types of elements, a wider range of calendars can be specified. Moreover, different conceptions can be used for them, generating multiple homogeneous views of the global annotated network. In this way, the calendar flexibility can be increased, and the entities that someone is interested in representing can be specified by means of a timed tabular view.

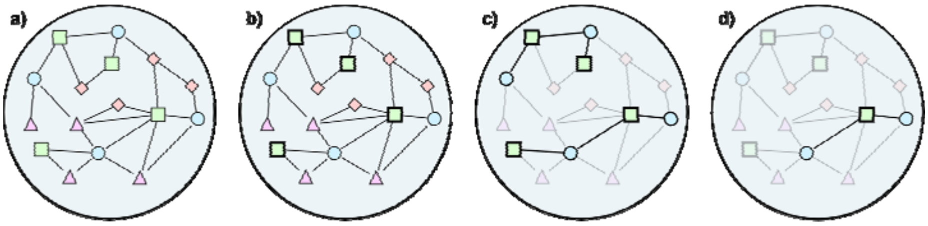

Figure 5: Construction of a calendar as a series of queries: a) the full semantic network; b) the semantic network with squares for the source entities (for example, the squares corresponding to actors are C^{actors}); c) target established are represented as circles entities (for example, if circles correspond to tasks, they are $C^{actors}{}_{tasks}$); and d) individual specific square chosen as the source (for example, if the chosen square is Mairi, it is $C^{actors\rightarrow\{name=Mairi\}\ tasks}$).

4 Prototype

This section provides a general description of the first version of the prototype for the proposal described in this research. Firstly, we describe the design of a suitable scenario where actors interact within. Secondly, we present a brief analysis of the feasibility of the proposal on the basis of this prototype under development. Although it is very initial studio, the aim is to provide an informative approximation to the future behaviour of the system. Finally, some technical and relevant information about the user interface implementation is described, in particular, the web-based navigation support for the calendar execution.

4.1 The scenario: an active lab

First of all, our work has been based upon the creation of an active space where users can freely interact using smart technology and the underlying middleware layer to manage hardware devices.

In particular, an active space was deployed at Universidad Autónoma de Madrid. This active space consists of a laboratory [UCAT] composed by a set of heterogeneous devices [Arroyo et al. 06]. A distributed blackboard architecture [Haya et al. 04] using Java technology, ICE middleware and MySQL database, harmonizes the interaction between the components of this room. Besides, this model combines raw sensor (presence, temperature, luminosity, etc.) and actuators (relays, dimmable light, etc.) using EIB, Phidget and X10 bus connections. For the multimedia information flow (images, digital radio, ip-camera, etc.) an Ethernet network is used. Each device has access to the context blackboard through the corresponding network(s).

User's identity and location are provided by means of two different solutions. Both mechanisms provide location at room level. On one the hand, RFID technology is used. Each user is univocally identified with a smart card. Therefore, every time a user enters the smart space, a RFID antenna reads his identity and updates his/her location. On the other hand, user's personal computer is equipped with activity sensor that allows knowing if a user is in his/her office. Location changes are propagated to the semantic network, and consequently, the corresponding calendars are notified as well.

The laboratory has been used as a seminar room. Actuators can be used as mediators between calendar events and the corresponding actions in the real world. For example, the lecture beginning or ending can be easily transformed from the calendar model to a notification using resources such as a LCD or a lamp. Furthermore, as we have shown, data sensor, such as RFID tag, can be converted into semantic information. Temperature, humidity or lighting conditions are included into to the room hypermedia model easing the user to check the laboratory status before each session.

Beside domotic devices, the laboratory contains a size flat screen for slide presentations. An API accessible through the blackboard allows external applications to upload and control the slides. The same blackboard facilitates conflict resolution when two or more different applications are trying to access the same resource. In particular, this is applied to the slide controller. It is very straightforward to integrate new modules that interact with the environment resources. We have showed that developing several context-aware applications from access control to user interfaces [Haya et al. 04]. In this respect, and going back to the previous slide's example, the speaker can control the presentation via voice [Montoro et al. 04] or via a graphical interface. These slides themselves are hypermedia information that is tied to the semantic model when a new seminar event is created. Then, the slides will be automatically available at the beginning of the session, and can be downloaded from the hypermedia network by any of the participants.

The distributed repository is the basic support for the hypermedia network. High-level queries filter this information according to date/time constrains and task requirements. This layer is implemented as a Web Service using Python, XML and MySQL. The resulting query is sent to Calendar layer in a readable form.

4.2 Feasibility studio

An important issue is the performance measurement of adaptive techniques in active spaces. The problem arise when hundreds of events has to been processed (user in/out event, user's task, user's calendar) in real time prior to giving an appropriated feedback. Therefore, we have analysed the performance of high intensive semantic network traversal to find useful information for a social aware community. Below, we explain the relationships of a given task (*GiveClass* Task) with the underlying semantic model.

The detailed behaviour of the prototype is illustrated as follows. When the user adds a *GiveClass* task to the system, the system creates a new subactivity entity and the corresponding set of relationships between the new element and the existing ones. Figure 6 shows a scheme reflecting the alterations in the system. Note that every element has a strong interrelation with the rest of elements.

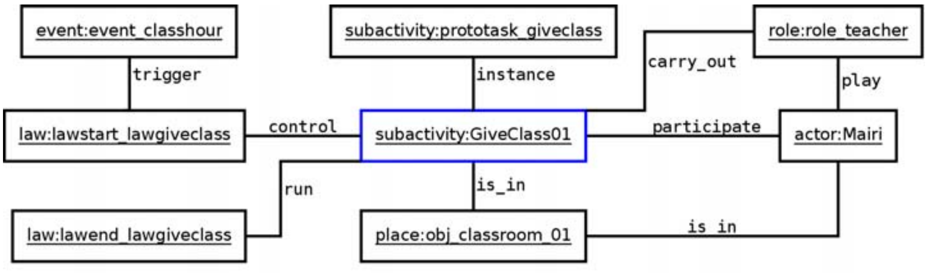

Figure 6: New entity (in blue) and relations created after adding a GiveClass element to the system.

Navigation through the semantic network must be done every time a query is triggered. For class calendars, queries imply information retrieval since the timetable is shared by all the students attending to the same class. This metaquery implies a set of specific queries, for example:

1. Fetch all the actors playing the role student for a specific class
2. For every actor, fetch all his/her tasks between two dates
3. Add every task to a single calendar
4. Return the calendar

So, the number of DB queries will grow directly with time intervals and actors implied. As a simple analysis of scalability, we provide some time estimations for query executions. First of all, we define the available valid timetable as five days per week (from Monday to Friday) and from 8:00 to 22:00. For simulation purposes, we will assume that every subject has 6 hours of theory plus laboratory per week, and 4 groups per subject, with 70 students per group. Thus, we can estimate the number of subactivity elements and the number of actors, and the relations between them.

MySQL defines the numbers of disk seeking [MySQL] as log(row_count) / log(index_block_length / 3 × 2 / (index_length + data_pointer_length)) + 1, being index block length usually 1.024 bytes, data pointer 4 bytes, and an index length of 4 bytes. This implies that we must attend to the number of queries and the size of tables. Performance will radically decrease if table index is greater than memory. The table index size can be calculated as row_count × 8 × 3/2, assuming a typical index buffer fill ratio of 2/3. Table 1 shows table rows, number of disk seeks and the table index size in Megabytes.

#subjects	#row_count	#disk_seek	index_size	#relations
1	29,592	3.32	0.34	29,568
2	118,320	3.63	1.35	118,272
3	266,184	3.81	3.05	266,112
4	473,184	3.94	5.42	473,088
5	739,320	4.04	8.46	739,200
6	1,064,592	4.12	12.18	1,064,448
7	1,449,000	4.19	16.58	1,448,832
8	1,892,544	4.25	21.66	1,892,352
9	2,395,224	4.30	27.41	2,395,008
10	2,957,040	4.35	33.84	2,956,800
11	3,577,992	4.39	40.95	3,577,728
12	4,258,080	4.43	48.73	4,257,792
13	4,997,304	4.47	57.19	4,996,992
14	5,795,664	4.50	66.33	5,795,328
15	6,653,160	4.53	76.14	6,652,800
16	7,569,792	4.56	86.63	7,569,408

Table 1: Growing of table rows, number of disk seeks, size index, and number of relations in function of number of subjects in the semantic network.

As the Table 1 shows, while the row count and table index grow up to 255 times, the disk seeks only increases 1,4 times, and the table index size remains lower than 128MB, which is not significant nowadays.

4.3 User interface

Regarding the user interface, an initial prototype of the calendar view interface is being developed in Mozilla Firefox browser. The advantages of Mozilla are cross-sourcing, cross-platform code, support for extensions and integration of different programming languages. Awareness is incorporated by using emerging toolbars (which can be opened and closed), added as a new information service transverse to any application running on the browser. Initially, two different bars have been defined: the *context bar* and the *radar bar*, as shown in Figure 8. The *context bar*, placed at the right side of the interface, is linked to the calendar, and provides access to constantly updated information reflecting the changes produced in real time. The interface shows a calendar that allows us to change the time interval to be considered, and three tabs to show related information: role and related-users location (users sharing a common task), activity, availability, and other relevant information (enabled communication channels, etc.) The second bar, the *radar-group bar*, is placed at the bottom. This bar is responsible of the user awareness, showing the status of the users who one has common tasks with at a certain moment. The browser main area constitutes the place where the hypermedia information is shown.

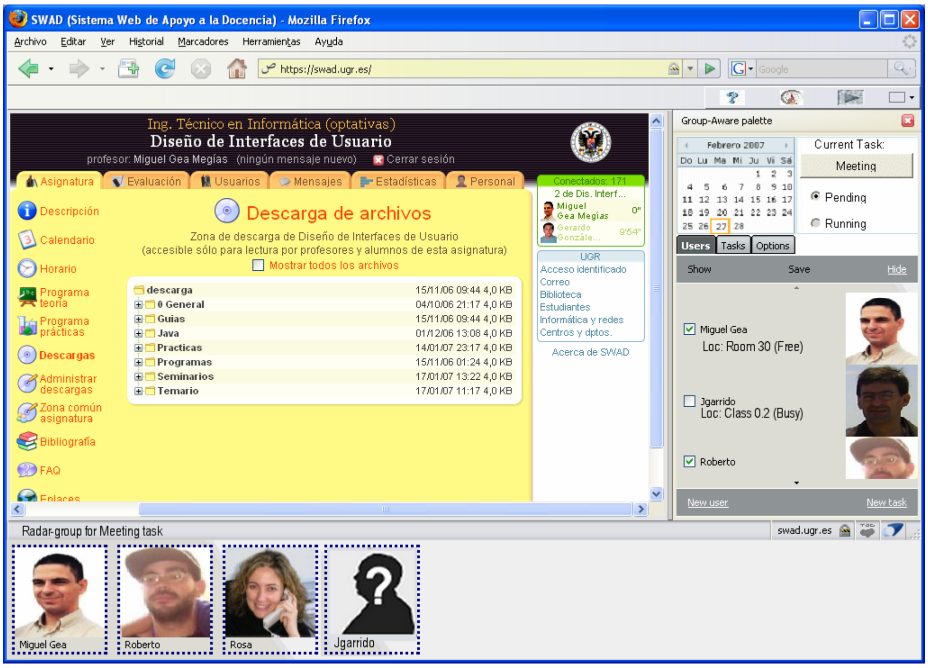

Figure 8: Snapshot of the calendar-based user interface: the group radar bar is shown at the bottom, whilst the context bar is shown on the right-hand side of the interface.

The user interface has been developed using XUL [Goodger et al. 01], a cross-platform graphical toolkit supporting XML and CSS. The hypermedia contents are rendered by the Web engine, Gecko. Communication with the semantic network is provided using the network component (Necko) and a simple cross-platform component model with multiple language bindings and IDL (XPCOM). This technology supports the Mozilla framework, which allows us to integrate the smart group-aware navigation system within the corresponding browser. The Gecko engine lets us to quickly show visual information, using XUL for GUI design. XML and CSS allow easy adaptability and customization. The Necko engine allows us to do network operations between layers, and the cross-platform component model facilitates specific language or platform independence.

5 Conclusions and Future Work

This paper presents a social-aware approach on active spaces using a smart calendar metaphor. This approach connects context-aware information obtained from active spaces to a social task model, facilitating the management of these concepts as a natural extension of adaptive tasks on ubiquitous environments.

Regarding the representation of complex underlying information, hypermedia constitutes a user-friendly interface which can be adapted at runtime according to user needs. A relevant issue in this work is the tight mapping of physical world to a hypermedia model, helping in the creation of tasks enhancing smart social awareness, as it has been exemplified in the b-learning scenario described above. Physical world and relevant concepts (roles, tasks and artefacts) are mapped on a hypermedia model which is traversed by a calendar allowing changes in both directions.

This approach represents a promising issue to authoring group-aware and context-dependent scenarios. We are developing a prototype using the Mozilla framework, representing user tasks and context information about current activities as well as the status of users involved. This system automatically generates adapted hypermedia interfaces for each user according to the role played at each time, providing dynamic task scheduling and decision support. Navigation on the underlying hypermedia model provides social aware support connecting physical world with planned tasks, facilitating the coordination of different users doing collaborative activities from diverse locations though different devices. The system implementation is based on a portable and extensible framework.

Calendars turn out to be particularly useful for scheduling activities in complex environments in which different types of users, devices, activities and contexts produce changes that must be managed. The integration of the calendar metaphor with hypermedia models and semantic networks contributes to the creation of context-aware collaborative environments.

Additionally, the framework described in this paper might be particularly useful in assistive environments for people with disabilities, such children with autism. We are developing a prototype for aiding children with severe communication impairments. Children will schedule activities through the calendar interface. These activities could be location-sensible (classroom, home, playground…) and imply different actors (teacher, monitors, relatives…). Children’s desires will be reflected on scheduled activities, and people will show the fulfilment of these desires through the progress and status of these activities.

Future work focuses on testing the system in a more extensive scenario that includes more dependencies and constrains. We plan to do a formal study with groups of teachers and students using the system, addressing usability and utility issues. The integration and synchronization of this model with other calendar systems (such as Google calendar or Sunbird) will also be explored, so that those calendars can be linked to physical entities. Another objective is to take the feasibility study a step further by considering other relevant non-functional requirements besides scalability, as mentioned in Section 2.3, in particular those of efficiency in the notification mechanisms and privacy.

Acknowledgements

This research has been partially supported by the Spanish Ministry of Science and Education, project TIN2007-64718.

References

[Abowd 99] Abowd, G. D. : "Classroom 2000: An Experiment with the Instrumentation of a Living Educational Environment"; IBM Systems Journal, Special issue on Pervasive Computing, Vol. 38-4 (1999), 508-530.

[Andueza and Carro 06] Andueza, N., Carro. R.M.: "Supporting Configurable Alerting Services in Mobile Learning Environments"; Procs. of the 8th Intl. Symposium on Information and Communication Technologies applied to Education, vol. 1. Endorsed by: IEEE Technical Committee on Learning Technology (ISBN: 84-9773-302-9), October (2006), 326-333.

[Arroyo et al. 06] Arroyo, R. F., Gea, M., Garrido, J. L., Haya, P. A.: "A Task-Driven Design Model for Collaborative AmI Systems"; Thibaud Latour and Michaël Petit, editors, Procs. CAiSE'06 Workshops and Doctoral Consortium. Presses Universitaires de Namur. Workshop: UMICS'06. (2006), 969-983.

[Bardram and Hansen 04] Bardram, J. E., Hansen, T. R.: "The AWARE Architecture: Supporting Context Mediated Social Awareness"; Mobile Cooperation Procs. of the 2004 ACM conference on Computer Supported Cooperative Work. ACM Press (2004), 192-201.

[Beard et al. 90] Beard, D., Palaniappan, M., Humm, A., Banks, D., Nair, A., Shan, Y. : "A visual calendar for scheduling group meetings"; Procs. of the 1990 ACM Conference on Computer-Supported Cooperative Work (Los Angeles, CA, USA, October 07-10, 1990). ACM Press, New York (1990), 279-290.

[Boyle 03] Boyle, M.: "A Shared Vocabulary for Privacy"; in Workshop on Ubicomp Communities: Privacy as Boundary Negotiation. Held as part of the UBICOMP'2003 5th Intl. Conference on Ubiquitous Computing, Seattle, October 12, (2003).

[Brzozowski et al. 06] Brzozowski, M., Carattini,. K., Klemmer,. S. R., Mihelich,. P., Hu, J., Ng, A. Y.: "groupTime: preference based group scheduling"; SIGCHI conference on Human Factors in computing systems CHI '06 . ACM Press (2006), 1047-1056

[Diaper 02] Diaper, D.: "Scenarios and task analysis"; Interacting with Computers, Vol. 14-4 (2002), 379-395.

[Carro 99] Carro, R.M., Pulido, E., Rodríguez, P.: "Dynamic generation of adaptive internet-based courses". Journal of Network and Computer Applications, vol. 22. Academic Press. (1999), 249-257.

[Dourish and Bellotti 92] Dourish, P., Bellotti, V.: "Awareness and Coordination in Shared Workspaces"; Proc. CSCW'92, ACM Press. Toronto, Canada (1992), 107-114.

[Ducatel et al. 01] Ducatel, K., Bogdanowicz, M., Scapolo, F., Leitjen, J., Burgelman, J. C.; "Scenarios for Ambient Intelligence in 2010"; Tech. Report, Information Society Technologies Advisory Group (ISTAG), Inst. of Prospective Technological Studies (IPTS), (2001).

[Erickson and Kellogg 00] Erickson, T., Kellogg, W.A.: "Social translucence: an approach to designing systems that support social processes"; ACM Transactions on Computer-Human Interaction, 7, 1 (2000) 59-83.

[Garrido et al. 05] Garrido J.L., Gea, M., Rodríguez, M.L.: "Requirements Engineering in Cooperative Systems"; Requirements Engineering for Socio-Technical Systems. Chapter XIV. IDEA GROUP INC. USA (2005), 226-244.

[Goodger et al. 01] Goodger, B., Hickson, I., Hyatt, D., Waterson, C.: "XML User Interface Language (XUL) 1.0"; http://www.mozilla.org/projects/xul/xul.html

[Gutwin and Greenberg 04] Gutwin, C., Greenberg, S.: "The importance of awareness for team cognition in distributed collaboration"; E. Salas and S.M. Fiore (Eds). Team Cognition: Understanding the Factors That Drive Process and Performance, Washington: APA Press, (2004), 177-201.

[Harris and Stocker 98] Harris, J. W., Stocker, H.: "Handbook of Mathematics and Computational Science"; New York, Springer (1998).

[Haya et al. 04] Haya, P. A., Montoro, G., Alamán, X.: "A prototype of a context-based architecture for intelligent home environments". LNCS, Vol. 3290. (2004), 477-491.

[Kutar et al. 00] Kutar M., Britton C., Nehaniv, C. L.: "Specifying Multiple Time Granularities in Interactive Systems"; Procs. DSV-IS 2000, LNCS, Vol. 1946. Limerick, Ireland, June (2000), 51-63.

[Mackinlay et al. 94] Mackinlay, J. D., Robertson, G. G., DeLine, R.: "Developing calendar visualizers for the information visualizer"; Procs. of the 7th Annual ACM Symposium on User interface Software and Technology (Marina del Rey, CA, USA, November 02 - 04, 1994). ACM Press, New York (1994), 109-118.

[Martin et al. 06] Martín, E., Carro, R.M., Rodríguez, P.: "A Mechanism to Support Context-based Adaptation in M-Learning"; Innovative Approaches for Learning and Knowledge Sharing. Nejdl, W. and Tochtermann, K. (Eds.), LNCS, Vol. 4227, (2006), 302-315.

[Montoro et al. 04] Montoro, G., Alamán, X., Haya. P. A.: "Spoken interaction in intelligent environments: a working system"; Advances in Pervasive Computing. Alois Ferscha, Horst Hoertner and Gabriele Kotsis, Eds. Austrian Computer Society (OCG). ISBN 3-85403-176-9 (2004), 217-222.

[Montoro et al. 04b] Montoro, G., Haya, P. A., Alamán, X.: "Context adaptive interaction with an automatically created spoken interface for intelligent environments"; LNCS, Vol. 3283 (2004), 120-127.

[Neustaedter and Bernheim Brush 06] Neustaedter, C., Bernheim Brush., A. J.: "LINC-ing" the family: the participatory design of an inkable family calendar; Procs. of the SIGCHI Conference on Human Factors in Computing Systems (Montréal, Quebec, Canada, April 22 - 27, 2006). ACM Press, New York (2006), 141-150.

[Ning et al. 02] Ning, P., Wang, X. S., Jajodia, S.: "An Algebraic Representation of Calendars"; Annals of Mathematics and Artificial Intelligence, 36, 1-2 (2002), 5-38.

[Payne 93] Payne, S.: "Understanding Calendar Use"; Human-Computer Interaction, 8, 2 (1993), 83-100.

[UCAT] U-CAT project web page. http://orestes.ii.uam.es/ucat/

[Weiser 91] Weiser, M.: "The computer for the 21st Century"; Scientific American, September, (1991), 94-110.

[Werle and Jansson 02] Werle, P., Jansson C. G.: "Achieving Non-intrusive Notifications in a Ubiquitous Service Environment"; Procs. of RadioVetenskap och Kommunikation, RVK 02. Stockholm, Sweden (2002).