Smart Offices and Intelligent Decision Rooms

Carlos Ramos, Goreti Marreiros, Ricardo Santos, Carlos Filipe Freitas

1 Introduction

Nowadays computing technology research is focused on the development of Smart Environments. Following that line of thought several Smart Rooms projects were developed and their appliances are very diversified. The appliances include projects in the context of workplace or everyday living, entertainment, play and education. These appliances envisage to acquire and apply knowledge about the environment state in order to reason about it so as to define a desired state for its inhabitants and perform adaptation to these desires and therefore improving their involvement and satisfaction with that environment.

Such "intelligent" or "smart" environments and systems interact with human beings in a helpful, adaptive, active and unobtrusive way. They may still be pro-active, acting autonomously, anticipate the users needs, and in some circumstances they may even replace the user. Decision Making is one of the noblest activities of the human being. Most times, decisions are taken involving several persons (group decision making) in specific spaces (e.g. meeting rooms). On the other hand, the

Carlos Ramos
GECAD and ISEP, e-mail: csr@dei.isep.ipp.pt

Goreti Marreiros
GECAD and ISEP, e-mail: goreti@dei.isep.ipp.pt

Ricardo Santos
GECAD and ESTGF, e-mail: rjs@estgf.ipp.pt

Carlos Filipe Freitas
GECAD , e-mail: cff@isep.ipp.pt

ISEP - Institute of Engineering of Porto, R. Dr. Antonio Bernardino de Almeida, 431, 4200-072 Porto, Portugal
GECAD - Knowledge Engineering and Decision Support Research Center at ISEP
ESTGF - School of Management and Technology at Felgueiras, Rua do Curral - Casa do Curral - Margaride - 4610 156 Felgueiras, Portugal

H. Nakashima et al. (eds.), *Handbook of Ambient Intelligence and Smart Environments*,
DOI 10.1007/978-0-387-93808-0_32,

present economic scenario (e.g. increased business opportunities, intense competition) needs fast decision-making to succeed.

In this chapter we will focus on smart offices and intelligent meeting rooms context. These topics are well studied and they intend to support the Decision Making activity, however, they have received a new perspective from the Ambient Intelligence concept, a different way to look at traditional offices and decision rooms, where it is expected that these environments support their inhabitants on a smart way, promoting an easy management, efficient actions and, most importantly, to support the creation and selection of the most advantageous decisions.

The topic in discussion, the Smart Offices appliances, are presented on the next section (2) where we include an overview of existent approaches and match them with definitions of such environments, followed with a detailed description of a few important projects present in literature, detailing in the hardware, software and capabilities perspectives. In section 3 we will discuss the Intelligent Meeting Rooms (IMR) where we will present a definition of such environments, as well as, some projects. On section 4 we will detail a Intelligent Decision Room project, LAID (Laboratory of Ambient Intelligence for Decision Making), installed at GECAD research center in ISEP. The environment created at LAID provides support to meeting room participants, assisting them in the argumentation and decision making processes, combining *rational* and *emotional* aspects. Special attention will be given to the ubiquity question, because in group decision context it is natural that group members are not available to meet at the same time and in the same space. We finish the document with some conclusions.

2 Smart Offices

To build intelligent environments it is vital to design and make effective use of physical components such as sensors, controllers and smart devices. Then, using the information collected by these sensors, the software e.g., intelligent agents, can reason about the environment and trigger actions in order to change the state of the environment, by means of actuators. Such sensors/actuators networks need to be robust and self-organized in order to create an ubiquitous/pervasive computing platform. The IEEE 1451 studies formalised the notion of a smart sensor as one that provides additional functions beyond the sensed quantity, such as signal condition or processing, decision-making functions or alarm functions [12]. This led to devices that do not intend to solve the entire intelligent environment problem [7]. However, they provide intelligent features confining to a single object and task.

The design of these sensors/actuators leads us to some pervasive computing and middleware issues. Here we can find challenges as invisibility, service discovery, interoperability and heterogeneity, pro-activity, mobility, privacy, security and trust [53]. In this way, to design and build a Smart Office, it becomes fundamental a selection of middleware that decreases the development and usability effort of software solutions that bring together all the data collected by the sensors, and then reasons about the environment and acts on it.

Now we will focus on the desired features for this peculiar AmI environments, the smart offices, present in literature.

In the "SENSE-R-US" project [36] researchers are focused in gathering real world data of employees in an office environment, identifying the steps to clear up that data and extract relevant information, in order to be able to build a meeting driven movement model for office scenarios. This research enables the produced system to be queried about the position and status of persons and to resolve queries such as "What is the temperature in room X?", "Which rooms are available?", "Is person X in a meeting?" or "How many meetings has person P already had today?".

On the other hand, in [34], the research is focused on using software agents to automate the environment customisation, enabling it to adapt to users' needs and to provide customised interfaces to the services available at each moment, without discarding security issues. It is expected a flexible support to users, devices and services addition and removal, scalability, reliable authentication and access control and to support devices with different capabilities, in terms of power supply, bandwidth and computer power.

In [19] we can find several approaches to solve the decision making task, defined as the process of determining the action that should be taken by the system in a given situation in order to optimise a given performance of one or more metrics. The algorithms establish a mapping from the gathered information at past observations with the known information about the current state of the environment and the inhabitants to choose a decision to be made at a current point in time. Decision making task problems are well documented in literature and can be summarised in three kinds of algorithms, they are *Reactive and Rule-Based Decision Systems*, *Planning Algorithms* and *Learning Algorithms*. In *Reactive and Rule-Based Decision Systems* decisions are based on a set of facts and rules that represent the various states of the environment and its inhabitants, joined at a set of condition-action maps. A rule is triggered when all of its conditions are satisfied. An example of such application can be seen in [24]. Inside of this approach different implementations are presented, such as Fuzzy Logic in [64] or influence diagrams when uncertainty is present in [50]. However this technique must be constructed by the programmer or the user, a problem that does not exist in *Planning Algorithms* which provides a framework that allows decision making agents to autonomously select the best actions. To perform these actions such systems have a model of the environment and the available actions and the resulting effects and are able to generate a solution in form of a sequence of actions that lead to the desired tasks' objectives. Several approaches of planning algorithms are exposed and discussed in detail in [50]. But both previous mechanisms do not provide the possibility of self customisation. This is only possible if the system designer or the user build customised models. The autonomous adaptation to the inhabitants by the system is only possible with *machine learning methods* such as Supervised Learning, however such techniques require the inhabitant or the programmer to provide a training set of data to the system. Such approaches are well reported by [37]. neural networks, Bayesian networks, Markov Models, Hidden Markov Models and State Predictors are others techniques that were already used, [9, 45, 46], in AmI features systems and these can dismiss the need of user or programmer intervention. Such algorithms pose many challenges in terms of safety considerations and computational efficiency mainly in large scale environments but the reverse of this medal is the environment possibility to perform many useful tasks and improve the comfort and energy efficiency. To allow highly complex environments to be controlled according to the preferences of its inhabitants, solutions pass by hierarchical multi-agents systems with reinforcement learning. Details of such techniques can be seen in [14, 3, 19, 6].

Besides the monitoring, prediction, autonomy and adaptation features already referred as key features of smart offices, [7] also focus on the capabilities of these environments to communicate with humans in a natural way. Normally classical offices use the commons Graphical User Interfaces (GUI) for interactions between user and system. However some authors consider that this kind of interfaces are intrusive and not truly user friendly, and to prove this theory a new class of interfaces emerged from the work on Human-Machine interaction. They are the Perceptual User Interfaces (PUI) that are not based on a keyboard and mouse couple as input interface but based on human actions like speech, gesture, interaction with object, etc. Very good examples of this kind of interfaces are Wellner's MagicDesk [61], Berard's MagicBoard in [4] detecting user commands, such as print or solve equations, introduced by human gestures and more recently the Intelligent Room Project [5] that is able to recognise human speech and gestures.

Now that we exposed common features present in such environments, we are ready to explore the definitions of smart environments present in literature. [25] defines a smart office as an environment that is able to help its inhabitants to perform everyday tasks by automating some of them and making the communication between user and machine simpler and effective.

In [33, 34] smart offices are defined as an environment that is able to adapt itself to the user needs, release the users from routine tasks they should perform, to change the environment to suit to their preferences and to access services available at each moment by customised interfaces.

In other words, Smart Offices contribute to reduce the decision-cycle offering, for instance, connectivity where-ever the user is, aggregating the knowledge and information sources. Smart offices handle several devices that support everyday tasks. Smart offices may anticipate user intentions, doing tasks on his behalf, facilitating other tasks, etc.

As we can see, the described definitions meet the features already presented. Now that we state the definition of Smart Office we will present what can be done in such environments, for that, on the next subsections we will report some important projects present in the literature, specifying the capabilities of the systems in terms of hardware, software and functionalities.

2.1 Active Badge

This is considered the first smart office application and its goal is to forward incoming calls for a user of a building to the nearest phone. [59]

The hardware consists on a badge, that the user must wear when he is inside of the building and emits unique infrared waves for each user, infrared sensors that receive the badge waves and a central database that is modified by the sensors when they contact with one badge.

The software consists on an application that is used by the receptionist to query the database about the last known position of a user. Then she/he is able to transfer the call to the telephone terminal nearest to the user.

Despite the simplicity and specificity of the system, the intermediate database enables the possibility of new sensors addition, even different kinds of mechanisms(e.g., like fingertips lockers), without the need to change or even restart the application.

2.2 Monica SmartOffice

In 1999 started the development of a smart office entitled of Monica [26, 25] that intended to anticipate user intentions and augment the environment to communicate useful information, through user monitoring.

In terms of hardware, this project involves 50 sensors (cameras and microphones) 4 mobile cameras, located in the corners, track one or more users, a wide-angle camera that observes the entire office and enables recognition of user activities and monitors the tracking strategy. A mobile camera on top of the users computer monitor permits a close-up view of the user at the computer, ideal for face recognition or facial-expression recognition. Microphones distributed across the ceiling are used for speech recognition. In terms of actuators, exist three, a video projector and two speakers. Finally six Linux Pentium III PCs connected with a fast Ethernet local network power Monica. Four computers perform vision processing for user tracking and activities recognition. A fifth PC is dedicated to the MagicBoard for both video projection and image processing. A sixth computer hosts the supervisor and is linked to the Internet.

Supporting this hardware 12 software modules exist, with specific tasks. They are the Gesture Recognition, a 3 dimensional mouse, a magic board and a magic desk, finger and click detection, a virtual assistance, MediaSpace for distributed collaborative work, Tracker, Face recognition, Activity detection, Phycons for user interaction and an Internet Agent and Gamma architectures witch allow the interrelation between all the others software agents.

2.3 Stanford's Interactive Worksapaces

The Stanford Interactive Workspaces project is a research facility called the iRoom, located in the Gates Information Sciences Building and focus on augmenting a dedicated meeting space with technology such as large displays, wireless/multimodal I/O devices, and seamless integration of mobile and wireless "appliances" including handheld PC's. The iRoom contains [20] three touch sensitive white-board displays along the side wall and a custom-built 9 megapixel, 6-foot diagonal display called the interactive mural. In addition, there is a table with a 3 x 4 foot display that was designed to look like a standard conference-room table. The room also has cameras, microphones, wireless LAN support, and several wireless buttons and other interaction devices.

The system infrastructure present in [20] is called the Interactive Room Operating System (iROS). It is a metaoperating system or middleware infrastructure tying together devices that have their own low-level operating system. iROS is composed by sub-systems named *Data Heap*, *iCrafter*, and the *Event Heap*. They are designed to address the three user modalities of moving data, moving control and dynamic application coordination, respectively. The inter-dependency between systems is very low, making the overall system more robust, and provides an easy deployment/replacement of devices.

2.4 Intelligent Environment Laboratory of IGD Rostock

This laboratory intended to create an interactive environment based on multimodal interfaces and goal-oriented interaction differing from other systems that normally use a function-oriented interaction. To do it, the components provide semantical interfaces with a description about the meaning and the effect of their functions in the environment. These semantic descriptions present in the Lab components take the form of a set of preconditions that must be fulfilled before an action can be triggered. The semantic description of the effects of that action provides the input to a planning assistant that is able to develop system comprehensive strategies and plans, even with new devices, to fulfil the goals [16].

The hardware includes a series of agents that control microphones, all kinds of sensors (temperature, brightness), standard devices and computer resources.

As software infrastructure the following modules are included: speech analysis and recognition, a dialogue management system (to translate user-goals into system goals), a context manager (for managing the system's view of the world), different sensors (to have perception of the environment), a planning assistant (to create strategies to fulfill the users goal), a scheduler (for plan execution), lots of devices (to have real-world effects).

This is a different approach to Smart Offices where the interactions on the Lab are goals to reach instead of actions to be performed.

2.5 Sens-R-Us

The Sensor-R-Us' application aims at building a Smart Environment in office context and is placed in the University of Stuttgart [36].

It has two types of sensor nodes, the "base stations" and the "personal sensors". *Base stations* are installed in all rooms, these serve as location beacons and gateways to the mobile nodes carried around by the employees and a PC based GUI, the position and status (e.g., 'in a meeting') of persons or sensor information like room temperature can be queried. *Personal sensors* receive the location beacons and determine their position and also send beacons which are received by other personal sensors to update their neighbourhood list. This list and microphone data is used to detect the occurrence of meetings.

2.6 Smart Doorplate

The Doorplate project is located at the University of Lancaster [57] and is able to display notes for visitors and change them remotely, e.g. by sending a text message (SMS) from the office owner. Thus in case of owner absence it should be able to perform simple secretary tasks, like accept/ forward messages, notify visitors of the current location of owner or return time.

Composed by a display, preferably a touch screen, and a microphone connected to a network, a person tracking system (identification tags and tag readers), additional sensors (to detect if a door is open) and actuators like electronic door locks and interconnection system software and middleware to connect the appliances are focused on the scenario of a visitor direction service, and is based on a peer-to-peer middleware architecture despite the commonly used client/server system.

2.7 Ambient Agoras

Researches made by [56] pursuit to augment the architectural envelope to create a social architectural space that supports collaboration, informal communication, and social awareness. The design of smart artifacts that users can interact with simply and intuitively in the overall environment were their main focus. For that, they introduce a distinction between two types of smart artifacts: *system-oriented, importunate smartness*, that creates an environment in which individual smart artifacts or the environment as a whole can take certain self directed actions based on previously collected information and leaves the human out from the decision process, and *people-oriented, empowering smartness* who places the empowering function in the foreground so that ˵smart spaces make people smarter ˝, empowering users to make decisions and take mature and responsible actions, putting the users in the decision loop.

The artifacts and software components developed include InfoRiver, InforMall, and the SIAM-system [47], Hello.Wall, ViewPort, and Personal Aura [56]. The *Hello.Wall* is an ambient display with 1.8 meter wide by 2 meter high which has integrated light cells and sensing technology and provides awareness and notifications to people passing by or watching. In a brief it is a piece of unobtrusive, calm technology that exploits people's ability to perceive information via codes. *ViewPorts* consists of a WLAN-equipped PDA-like handheld device with RFID readers and transponders so that a ViewPort can sense other artifacts and be sensed itself. It is used by Hello.Wall, which borrows the ViewPort's display to privately show more explicit and personal information that can be viewed only on a personal or temporarily personalized device. Depending on access rights and current context, people can use ViewPorts to learn more about the Hello.Wall, to decode visual codes on the wall, or to access a message announced by a code. *Personal Aura* is an easy and intuitive interface that let users control their appearance in a smart environment. They can decide whether to be visible to a tracking system and, if so, they could control the social role in which they appeared. The participants indicated [56] that they perceived the Hello.Wall as being an appropriate means of establishing awareness of people who were working at a remote site, thus overcoming the isolation of not being physically present without causing privacy problems. they also cite that the interactions with the remote site took place more often, spontaneous video conference interactions were less formal, and videoconferencing became a daily routine.

3 Intelligent Meeting Rooms

The increasing competitiveness present in the business world led people and organisations to take decisions in a short period of time, in a formal group setting and in specific spaces (e.g. meeting rooms) [41]. As those decisions have to be the most advantageous, taking into account the quality of the final results, the researchers have developed several decision support systems. At the beginning developed systems aimed to support face-to-face meetings [41]; today's the aim is to develop systems that support distributed and asynchronous meetings, naturally allowing a ubiquitous use that can add flexibility to the global organisational environment of today [13].

Such orientation in software development was followed by the design and building of rooms with specific hardware and software that could empower the decisions makers' actions, supporting them with knowledge and driving their attention to the problem and avoiding their minds from wandering on needless issues. This kind of rooms is commonly named by Intelligent Meeting Rooms (IMR), and can be considered as a sub domain of Smart Rooms for the workplace context.

The generic goal of Intelligent Meeting Rooms is normally referred as a system that supports multi-person interactions in the environment in real time, but also as a system that is able to remember the past, enabling review of past events and the reuse of past information in an intuitive, efficient and intelligent way [35]. Besides this classical definition [29] cite that IMR should also support the decision making process considering the emotional factors of the intervenient participants, as well as the argumentation process.

The infrastructure used for such rooms include a suite of multimodal sensory devices, appropriate computing and communications systems. A good identification of the components in a smart environment and their connection can be seen in [7].

The following subsections provide details on IMR projects present in literature focusing on their tasks, hardware and software capabilities. For now we will present the systems that fit in the [35] and on the next section we will refer a project that we have developed, the LAID project [32], that goes further then the classical definition of IMR and also meets the [29] definition.

3.1 The IDIAP Smart Meeting Room

In the field we can find interesting projects such as IDIAP smart meeting room [38] receiving meetings containing up to twelve participants. The hardware is composed by a table, whiteboard, computer projection screen, 24 microphones configured as lapel microphones, in the ears of a binaural manikin, and in a pair of 8 channel tabletop microphone arrays, three video cameras, and equipment for capturing time-stamped whiteboard strokes.

On IDIAP recorded data is precisely synchronized so that every microphone, pen-stroke, and video sample can be associated with simultaneously captured samples from other media streams. The software component uses XML to catalogue all the data and it is mentioned an off-line media interactive browsing system.

3.2 SMaRT

SMaRT [58] was developed at ISL (Interactive Systems Laboratories) and intends to provide meeting support services that do not require explicit human-computer interaction, enabling the room to react appropriately to users' needs maintaining the focus on their own goals. It supports human-machine, human-human, and human-computer-human interactions providing multi modal and fleximodal interfaces for multilingual, multicultural meetings. It is instrumented to collect data on behaviors so that the participants' interactions and their activities on the meeting can be analyzed.

3.3 M4 and AMI

Literature also refers the M4 (Multi Modal Meeting Manager) project as a large-scale project funded by the European Union in its 5th Framework Programme [40].

M4 aim is to design a meeting manager that is able to translate the information that is captured from microphones and cameras into annotated meeting minutes that allow for high-level retrieval questions, as well as summarization and browsing. It is concerned with the building of a demonstration system to enable structuring, browsing and querying of an archive of automatically analysed meetings.

The software goal of the M4 project includes the analysis and processing of audio and video streams, robust conversational speech recognition, to produce a word-level description, recognition of gestures and actions, multimodal identification of intent and emotion, multimodal person identification and source localization and tracking. This inferred data can be accessed from the M4 meeting manager.

In this project is also included AMI (Augmented Multi-party Interaction) software [40] concerned with new multi modal technologies to support human interaction, in the context of smart meeting rooms and remote meeting assistants. It aims to enhance the value of multi modal meeting recordings and to make human interaction more effective in real time. Its main focus is on using the hardware to 3D centroid tracking, person identification and current speaker recognition, event recognition for directing the attention of active cameras, best view camera selection, active camera control and in the Graphical summarization/user interface.

3.4 NIST Smart Space Project

The NIST Meeting Room [55] is an US Department of Commerce project that seeks pervasive devices, sensors, and networks, for context-aware smart meeting rooms that sense ongoing human activities and respond to them.

In terms of hardware this IMR has over two hundred microphones, five HD cameras, a smart whiteboard, and a locator system for the meeting attendees. Together, these sensors generate over a gigabyte per minute of sensor data, which are time tagged to millisecond resolution and stored. [11]

This project includes 2 major software toolkits, the ATLAS and SmartFlow. The sensors streams are managed by the *NIST SmartFlow* system, a data flow middleware layer that provides a data transport abstraction. NIST SmartFlow system generates the connections and transports the data among the clients and it consists of a defined middleware API for real-time data transport, and a connection server for sensor data source, processing, display, and storage graph node. Thus being SmartFlow layer makes possible to integrate components from multiple sources, such as speaker identification and speech recognition systems.

Architecture and Tools for Linguistic Analysis Systems (ATLAS) is a signal-independent, linguistic annotation framework that is responsible for the extraction of low-level semantic descriptions in the form of metadata, or annotations from the data streams and subsequent inferring of higher-level annotations about the users tasks on meeting context.

In the scope of this project some products already have been released to the market such as a NIST Microphone Array Series, designed to minimise analog signal paths to reduce noise, and also allowing flexible deployment in research laboratories. [10]

3.5 MIT Intelligent Room Project

The coordinator of this project, Brooks [5] have argued that Intelligent Environments (IE) should adapt to, and be useful for, ordinary everyday activities; they should assist the user, rather than requiring the user to attend to them; they should have a high degree of interactivity; and they should be able to understand the context in which people are trying to use them and behave appropriately.

The software environment supports change, it is an adaptable system. Change appears by means of anonymous devices customised to users, by explicit user requests, by the needs of applications and different operating conditions [54]. This project has produced several communications and on the following text we will cite some of them.

MIT's intelligent room prototype facilitates activities via two systems the *Rascal* and *ReBa*, which are responsible for the adaptive and reactive components and provides a way to make use of perceptual information. They developed a Semantic Network [21] a graph notation for representing the knowledge of the context in patterns of interacted nodes and links, which represents a model for capturing a vast amount of information from cameras, microphones, and sensors, for creating context-aware room reactions. There also exists *MetaGlue*, that can be seen as an execution platform for distributed agents having a dynamic nature that allows the inclusion of new agents in the system in run time, resource discovery and agent state recovering makes the agents "invincible". In terms of collaboration, *Meeting View* and *The eFacilitator* are supporting tools that make possible to effectively visualise the content of a meeting. [54]

SAM, *Look-To-Talk (LTT)* and *Multimodal Sketching* are human computer interfaces (HCI). *SAM* is an HCI project in which was developed an expressive and responsive user interface agent for intelligent rooms. *LTT* interface involves an animated avatar representing the room and it intends to engage in speech recognition capabilities of the environment into human activities. *Multimodal Sketching* aim is to integrate speech into the existing sketching system, which helps to get an idea of the user's intentions.

4 LAID - Laboratory of Ambient Intelligence for Decision Support

LAID (Laboratory of Ambient Intelligence for Decision Making) is an Intelligent Environment to support decision meetings (Fig. 1). The meeting room supports distributed and asynchronous meetings, so participants can participate in the meeting

Fig. 1 Ambient Intelligent Decision Lab

where ever they are. The software included is part of an ambient intelligence environment for decision making where networks of computers, information and services are shared [31]. In these applications emotions handling are also included. The middleware used is in line with the live participation/supporting on the meeting. It is also able to support a past review in an intuitive way, however the audio and video remember features are still in arrangement.

The GECAD Intelligent Decision Room is composed by the followed hardware:

- Interactive 61 plasma screen
- Interactive holographic screen
- Mimio Note grabber
- One interactive 26 LCD screen (Six of this) used by 3 decision points
- 3 cameras, Microphones and activating terminals controlled by a CAN network.

With this hardware we are able to gather all kind of data produced on the meeting, and facilitate their presentation to the participants, minimizing the middleware issues to the software solutions that intend to catalog, organize and distribute the meeting information.

As an example of a potential scenario, it is considered a distributed meeting involving people in different locations (some in a meeting room, others in their offices, possibly in different countries) with access to different devices (e.g. computers, PDAs, mobile phones, or even embedded systems as part of the meeting room or of their clothes). Fig. 1 shows an Ambient Intelligent Decision Laboratory with several interactive Smartboards. The meeting is distributed but it is also asynchronous, so participants do not need to be involved at any time (like the meeting participant using a PDA and/or a notebook in Fig. 2).

However, when interacting with the system, a meeting participant may wish to receive information as it appears. Meetings are important events where ideas are exposed, alternatives are considered, argumentation and negotiation take place, and where the emotional aspects of the participants are so important as the rational ones. This system will help participants, showing available information and knowledge, analysing the meeting trends and suggesting arguments to be exchanged with others.

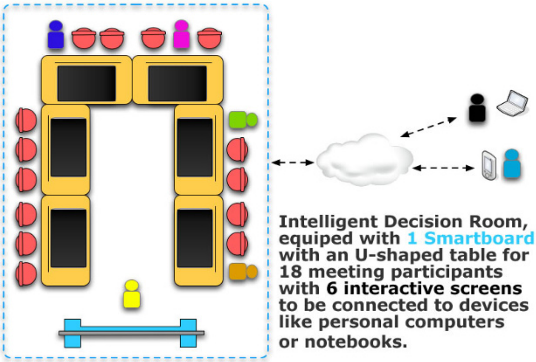

Fig. 2 Distributed Decision Meeting

4.1 Ubiquitous Group Decision Making

The term Group Decision Support System (GDSS) [17, 27] emerged effectively in the beginning of the eighty-decade. According to Huber [18] a GDSS consists of a set of software, hardware, languages components and procedures that support a group of people engaged in a decision related meeting. A more recent definition from Nunamaker and colleagues [42] says that GDSSs are interactive computer-based environment which support concerted and coordinated team effort towards completion of joint tasks. Generically we may say that GDSS aims to reduce the loss associated to group work (e.g. time consuming, high costs, improper use of group dynamics, etc.) and to maintain or improve the gains (e.g. groups are better to understand problems and in flaw detection, participants' different knowledge and processing skills allow results that could not be achieved individually). The use of GDSS allows groups to integrate the knowledge of all members into better decision making. Jonathan Grudin [15] classifies the digital technology to support the group interaction into three phases: the pre-ubiquitous, the proto-ubiquitous and the ubiquitous ones. In the pre-ubiquitous phase, that begun in the 70's, face-to-face meetings were supported. In the proto-ubiquitous phase, distributed meetings were supported. This phase came to life in the 90's. The ubiquitous phase is now getting under way, supports meetings, and it is distributed in time and space. ubiquitous computing was introduced in the 90's and anticipates a digital world which consists in many distributed devices that interact with users in a natural way [60]. This vision was too far ahead for its time, however the hardware to implement Mark Weiser's vision is now commercially available and at a low cost. In an ambient intelligent environment, people are surrounded with networks of embedded intelligent devices providing ubiquitous information, communication and services. Intelligent devices are available whenever needed, enabled by simple or effortless interactions, attuned to senses, adaptive to users and contexts, and acting autonomously. High quality information and content may therefore be available to any user, anywhere, at any time, and on any device. Today, there is an increasing interest in the development of ubiquitous group decision support systems (uGDSS) to formalise and develop "any time and any place" group decision making processes, instead of using the conventional "same place and same time" approach. This interest came with the need of joining the best potential group of participants. With the economy globalization, possible participants to form the group, like specialist or experts in specific domains, are located in different points of the world and there was no way to put them in the same decision room. Until some years ago, a way out of this scenario was to wait until all the participants meet together. Actually, there is a growing interest in developing systems to hold up such scenarios. There are many areas where ubiquitous group decision making apparently makes sense. One of the most cited areas in literature is healthcare, since patients' treatment involves several specialists, like physicians, nurses, laboratory assistants, radiologists. These specialists could be distributed along departments, hospitals or even living in different countries. The HERMES system, a web-based GDSS was tested according to this scenario [22]. There are other GDSS that support ubiquitous decision making (GroupSystems software; WebMeeting software; VisionQuest software).

4.2 Ubiquitous System Architecture

Here, our objective is to present a ubiquitous system able to exhibit an intelligent and emotional behavior in the interaction with individual persons and groups. This system supports persons in group decision making processes considering the emotional factors of the intervenient participants, as well as the argumentation process.

Groups and social systems are modeled by intelligent agents that will be simulated considering emotional aspects, to have an idea of possible trends in social/group inter-actions.

The system consists of a suite of applications as depicted in Fig. 3.

Its main goals of the system are [31]:

- The use of a simplified model of Groups and Social Systems for Decision Making processes, balancing Emotional and Rational aspects in a correct way;
- The use of a decision making simulation system to support meeting participants. This will involve the emotional component in the decision making process;
- The use of an argumentation support system, suggesting arguments to be used by a meeting participant in the interaction with other participants;
- The mixed initiative interface for the developed system;
- The availability of the system in order to be used in any place, in different devices and at different times.

The main blocks of the system are:

- WebMeeting Plus is an evolution of the WebMeeting project [28] with extended features for audio and video streaming. In its initial version, based on

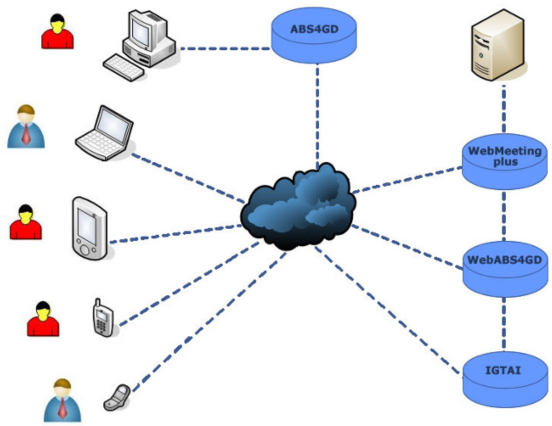

Fig. 3 System architecture

Web-Meeting, it was designed as a GDSS that supports distributed and asynchronous meetings through the Internet. The WebMeeting system is focused on multi-criteria problems, where there are several alternatives that are evaluated by various decision criteria. Moreover, the system is intended to provide support for the activities associated with the whole meeting life cycle, i.e. from the pre-meeting phase to the post-meeting phase. The system aims to support the activities of two distinct types of users: ordinary group "members" and the "facilitator". The system works by allowing participants to post arguments in favor/neutral/against the different alternatives being discussed to address a particular problem. It is also a window to the information repository for the current problem. This is a web based application accessible by desktop and mobile browsers and eventually WML for WAP browsers;

- ABS4GD (Agent Based Simulation for Group Decision) is a multi-agent simulator system whose aim is to simulate group decision making processes, considering emotional and argumentative factors of the participants [15]. ABS4GD is composed by several agents, but the more relevant are the participant agents that simulate the human beings of a decision meeting (this decision making process is influenced by the emotional state of the agents and by the exchanged arguments). The user maintains a database of participant's profiles and the group's model history; this model is built incrementally during the different interactions of the user with the system;
- WebABS4GD is a web version of the ABS4GD tool to be used by users with limited computational power (e.g. mobile phones) or users accessing the system through the Internet. The database of profiles and history will not be shared by all users, allowing for a user to securely store its data on the server database, which guarantees that his/her model will be available for him or her at any time;

- IGTAI (**I**dea **G**eneration **T**ool for **A**mbient **I**ntelligence) is an Idea Generation Tool to support a ubiquitous group decision meeting dedicated to the idea generation task. It is a tool designed to users with little experience in informatics systems, with group knowledge management, ubiquitous access, user adaptiveness and pro-activeness, platform independence and the formulation of a multi - criteria problems at the end of a work session.

4.3 ABS4GD and WebMeeting Plus

agent based simulation is considered an important tool in a broad range of areas e.g. individual decision making (``what if´´scenarios), e-commerce (to simulate the buyers and sellers behaviour), crisis situations (e.g. simulate fire combat), traffic simulation, military training, entertainment (e.g. movies).

According to the architecture that we are proposing we intend to give support to decision makers in both of the aspects identified by Zachary and Ryder [63], namely supporting them in a specific decision situation and giving them training facilities in order to acquire competencies and knowledge to be used in a real decision group meeting. We defend that agent based simulation can be used with success in both tasks. As referred in the introduction multi-agent systems seem to be quite suitable to simulate the behaviour of groups of people working together [29, 8]. Each participant of the group decision making process is associated with a set of agents to interact with other participants. The community should be persistent because it is necessary to have information about previous group decision making processes, focusing credibility, reputation and past behaviours of other participants [2].

There are three different types of agents in our model: Facilitator agent, Assistant agent and the Participant agent. The *Facilitator agent* helps the responsible for the meeting in its organization (e.g. decision problem and alternatives definition). During the meeting, the Facilitator agent will coordinate all the process and, at the end, will report to the responsible the results of the meeting. The *Assistant agent* works as an assistant to the participant of the meeting presenting all the updated information of the meeting. This agent acts like a bridge between the participant (user) and the participant agent. The *Participant agent* is the most important agent of the model and will be described with more detail in the next section.

4.3.1 Participant Agent

The participant agent represents a very important role in the group decision support system. For that reason, we will present the architecture and a detailed view of all the component parts. The architecture is divided into three layers that are: the knowledge layer, the communication layer and the reasoning layer as seen on Fig. 4.

In the *knowledge layer* the agent has information about the environment where he is situated, about the public profile of the other participant's agents that compose the decision meeting group, and regarding its own preferences and goals (its own public and private profile). The information in the knowledge layer is dotted of uncertainty [39] and will be accurate along the time through interactions done by the agent.

The *interaction layer* is responsible for the communication with other agents, the interface with the user of the group decision making simulator and by the mixed initiative interaction between participant and agents.

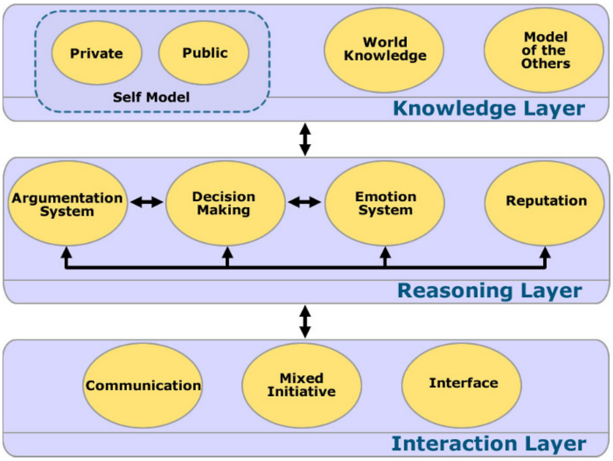

Fig. 4 Participant Agent Architecture

The *reasoning layer* contains three major components:

- *The argumentative system* that is responsible for the arguments generation. This component will generate explanatory arguments and persuasive arguments, which are more related with the internal agent's emotional state and about what he thinks of the others agents' profile (including the emotional state).
- *The decision making module* will support agents in the choice of the preferred alternative and will classify all the set of alternatives in three classes: preferred, indifferent and inadmissible;
- *The emotional system* [52] will generate emotions and moods, affecting the choice of the arguments to send to the others participants, the evaluation of the received arguments and the final decision.
- *The reputation module* support the user in the definition of the level of trust to the participant agent in the delegation of actions.

4.3.2 Argumentation System

Arguments may be classified according to type. Here we assume that the following six types of argument have persuasive force in human negotiations [43] [48] : threats; promise of a future reward and appeals; appeal to past reward; appeal to counter-example; appeal to prevailing practice; and appeal to self interest. These are the arguments that agents will use to persuade each other. This selection of arguments is compatible with the power relations identified in the political model: reward, coercive, referent, and legitimate [51]. This component will generate persuasive arguments based on the information that exists in the participant's agent knowledge base [30].

Argumentation Protocol

During a group decision meeting, participant agents may exchange the following locutions: request, refuse, accept, request with argument.

- Request (AgP_i, AgP_j, α, arg) - in this case agent AgP_i is asking agent AgP_j to perform action α, the parameter arg may be void and in that case it is a request without argument or may have one of the arguments specified in the end of this section.
- Accept (AgP_j, AgP_i, α) - in this case agent AgP_j is telling agent AgP_i that it accepts its request to perform α.
- Refuse (AgP_j, AgP_i, α) - in this case agent AgP_j is telling agent AgP_i that it cannot accept its request to perform α.

The purpose of the agent participant is to replace the user when he is not available. For example, in Fig. 5, it is possible to see the argumentation protocol for two agents. However, note that one of the participants is not available at the moment leaving all the actions to the participant agent (AgP_2). Note that this is the simplest scenario, because in reality, group decision making involves more than two agents and, at the same time that AgP_1 is trying to persuade AgP_2 that agent may be involved in other persuasion dialogues with other group members.

The autonomy of the participant agent is connected with the trust that the real participant has in the agent. As the agent exchanges locutions with other participants the user can approve or reject the locutions made by the participant agent there are no definitive locutions made by the agent. In that case trust level can be increased or decreased.

Arguments Selection

In our model it is proposed that the selection of arguments should be based on agent emotional state. We propose the following heuristic:

- If the agent is in a good mood he will start with a weak argument;
- If the agent is in bad mood he will start with a strong argument.

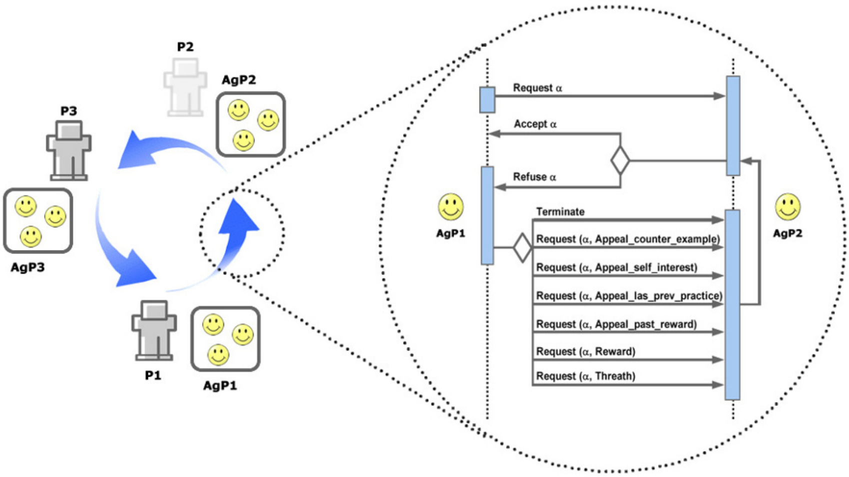

Fig. 5 Argumentation protocol

We adopt the scale proposed by Kraus for the definition of strong and weak arguments, where the appeals to a prevailing practice are the weakest and threats are the strongest arguments. We defined two distinct classes of arguments, namely a class for the weaker ones (i.e., appeals) and a class for the remainders (i.e., promises and threats). Inside each class the choice is conditionally defined by the existence in the opponent profile of a (un)preference by a specific argument. In case the agent does not detain information about that characteristic of the opponent, the selection inside each class follow the order defined by Kraus [23].

Arguments evaluation

In each argumentation round the participant agents may receive requests from several partners, and probably the majority is incompatible. The agent should analyse all the requests based on several factors, namely the proposal utility, the credibility of proponent and the strength of the argument. If the request does not contain an argument, the acceptance is conditioned by the utility of the request for the self, the credibility of the proponent and one of its profile characteristics, i.e., benevolence.

We consider:

$$\mathrm{Req}^t_{AgP_i} = \{Request^t_1(AgP, AgP_i, Action), \ldots, Request^t_n(AgP, AgP_i, Action)\},$$

where AgP represents the identity of the agent that perform the request, n is the total number of requests received at instant t and Action the request action (e.g., voting on alternative number 1). The algorithm for the evaluation of this type of requests (without arguments) is presented next:

if $\neg profile_{AgP_i}(benovolent)$ **then**
  foreach $request(Proponent, AgP_i, Action) \in Req^t_{AgP_i}$ **do**
    $refuse(Proponent, AgP_i, Action)$
  end
else
  foreach $request(Proponent, AgP_i, Action) \in Req^t_{AgP_i}$ **do**
    if $AgPO_{AgP_i} \vdash Action$ **then**
      $Requests \leftarrow Requests \bigcup request(Proponent, AgP_i, Action)$
    else
      $refuse(Proponent, AgP_i, Action)$
    end
  end
  $(AgP, RequestedAction) \leftarrow SelectedMoreCredible(Requests)$
  foreach $request(Proponent, AgP_i, Action) \in Requests$ **do**
    if $Proponent = AgP$ *or* $RequestedAction = Action$ **then**
      $accept(Proponent, AgP_i, Action)$
    else
      $refuse(Proponent, AgP_i, Action)$
    end
  end
end

If the requests are accomplished with arguments, then evaluation is performed using the following criteria: strength of the argument, opponent credibility, existence of preference for the argument, quality if the information detained about the opponent, convincing factor (analyze the validity of the argument).

4.3.3 Emotional System

The emotions that will be simulated in our system [29] are those identified in the reviewed version of the OCC model: joy, hope, relief, pride, gratitude, like, distress, fear, disappointment, remorse, anger and dislike. The agent emotional state (mood) is calculated in this module based on the emotions felt in the past and in others agents' mood [52]. Each participant agent has a model of the other agents, in particular has information about the other agent's mood. This model of the others considered incomplete information handling and the existence of explicit negation, following the approach described in [1].

Some of the properties that characterise the agent model are: gratitude debts, benevolent, credibility [2], (un)preferred arguments. Although the emotional component is based on the OCC model we think that with the inclusion of mood we can overcome one of the major criticisms that usually is pointed to this model, the fact that OCC model does not handle the treatment of past interactions and (in our case) past emotions.

4.4 Idea Generation Techniques

IGTAI aims to support the group in the idea generation task. So, in this section the study made is briefly exposed. Today several Idea Generation Techniques are known e.g the Nominal Group Technique (NGT), Theory of Inventive Problem Solving Professional Classes-Great Results (TRIZ), Mind-mapping, Brainstorming [44], co-operative KJ [62]. Mind-mapping at its most basic form is a simple hierarchy and could be drawn in any tree-shaped format. The idea is to add a formal structure to thinking, starting on the question that is intended to answer. Brainstorming is a problem resolution method which aims the generation of the maximum number of ideas in order to solve a problem in a collaborative and non-critical atmosphere. Alex Osborn, the founder of this method in 1938 cites that "in group a regular person can create two times or more ideas than in singular". However for the success of this method four rules and two principles must be followed. The two principles are the "judgment delay" and "the amount creates quality". The first principle means that in the use of this technique there is no space to criticise the other's ideas because that could stop the improvement of some ideas. The second one says that the existence of more ideas means a better final solution. In this technique exist four rules that must be followed and they are focus on quantity (the greater the number of ideas generated, the greater the chance of producing a radical and effective solution), no criticism (instead of immediately stating what might be wrong with an idea, the participants focus on extending or adding to it, reserving criticism for a later 'critical stage' of the process), unusual ideas are welcome (they may open new ways of thinking and provide better solutions than regular ideas) and finally combine and improve ideas (Good ideas can be combined to form a very good idea, this approach leads to better and more complete ideas than just generation of new ideas, and increases the generation of ideas, by a process of association).

4.5 Implementation

Some implementation details of the simulator (ABS4GD), the WebMeeting Plus and the idea generation tool (IGTAI) are described here.

ABS4GD

The ABS4GD is developed in Open Agent Architecture (OAA), Java and Prolog. OAA is structured in order to: minimize the effort involved in the creation of new agents, that can be written in different languages and operating on diverse platforms;

encourage the reuse of existing agents; and allow for dynamism and flexibility in the creation of agent communities. More information about OAA can be found in www.ai.sri.com/ oaa/. Some screens of the ABS4GD prototype running on LAID (Fig. 1) may be found in Fig. 7 and Fig. 6.

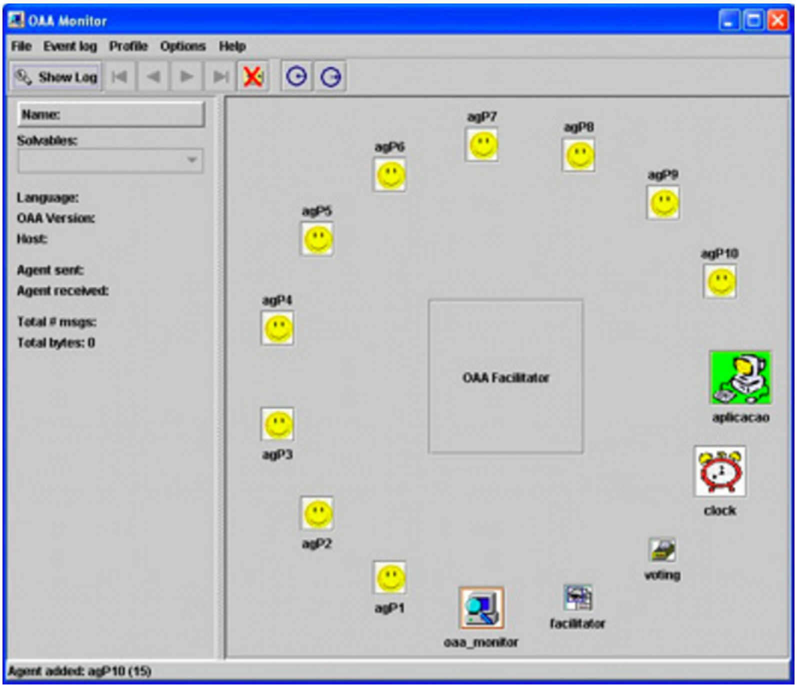

Fig. 6 Community of agents

Fig. 6 shows the collection of agents that work at a particular moment in the simulator: ten participant agents, the facilitator agent (responsible for the follow-up of all simulations), the voting agent, the clock agent (OAA is not specially designed for simulation, for that reason it was necessary to introduce a clock agent to control the simulation), the oaa_monitor (i.e. an agent that belongs to the OAA platform, and is used to trace, debug and profile communication events for an OAA agent community) and the application agent (responsible for the communication between the community of agents and the simulator interface).

Fig. 7 shows an extract of the arguments exchanged by the participant agents. Once a simulation is accomplished, agents update the knowledge about the other agents' profile (e.g. agent credibility).

WebMeeting Plus

The Webmeeting plus is a ubiquitous application intended to be used in a web

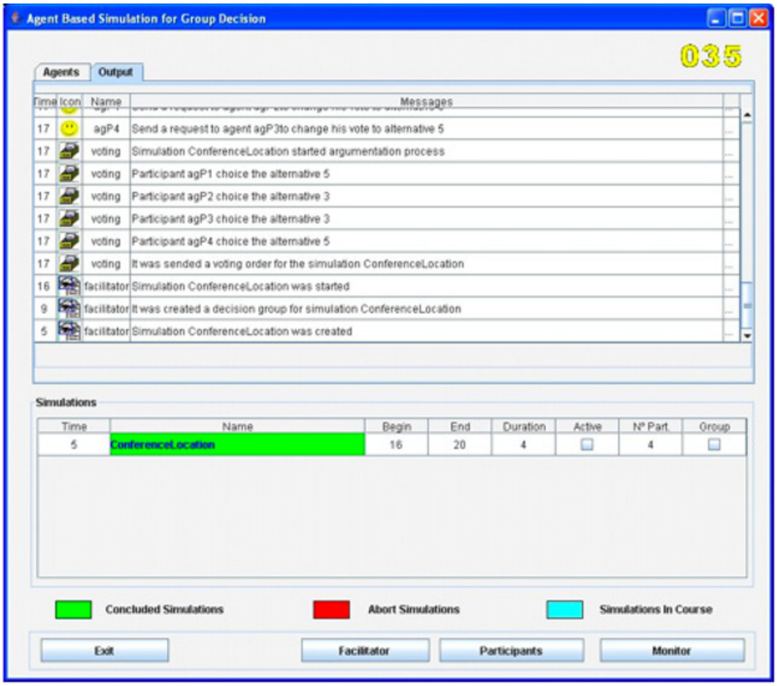

Fig. 7 Argumentation dialogues

browser and is developed in JavaScript, Java J2SE 5.0, OAA and Prolog. JavaScript and Java is used in the development of the main application, the OAA and the Prolog is used in the implementation of the agents. The participants, as well as the organizer, can use it in our Ambient Intelligence for Decision Support Lab (Fig. 1) or in any other computer system. The only requirement is to have Internet access. This system intends to reduce the disadvantages of traditional meetings, delegating all the time consuming and boring duties to the agents. All the participants of the meeting have an agent that personifies him in the interaction with other participants in the meeting. The main screen of the system can be seen in Fig. 8.

Fig. 9 on the left presents a window of the responsible of the meeting where it is possible to see, in the top, a list of all the participants in the meeting. The public profile of each one of the participants can be visualised by the responsible if he intends to do it. In the bottom of the window it is presented a graphic with the trend of the meeting. For each alternative the characteristics can be visualised, as well as how many participants may support that alternative. On the right there is a window of the participant where it is possible to see a table with all the requests made by a participant/agent of a meeting to all the other participants.

For each request there is a response made by the participant who received the request. For a participant, it is possible to change the requests made till then (e.g. he can change his mind about some subject or disagree with a request made by his

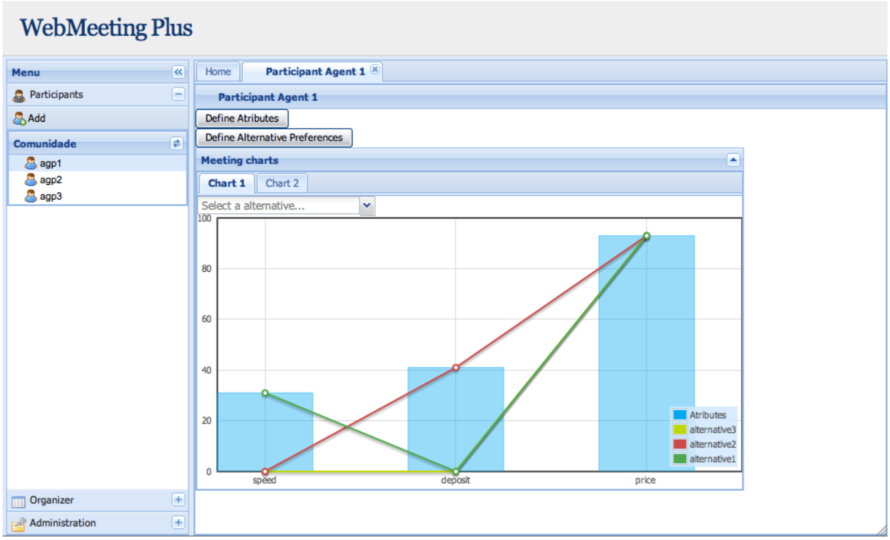

Fig. 8 WebMeeting Plus

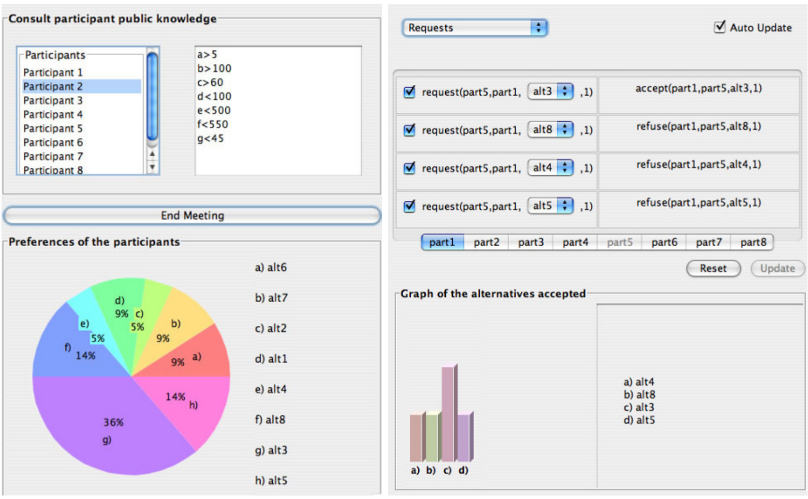

Fig. 9 Organizer and Participant Window

agent). In the bottom it is possible to see a graphic that shows all the preferred alternatives of the participant. The graphic shows to the participant the trends and the chances of each alternative.

IGTAI

The IGTAI was developed in Java J2SE 5.0 and the communication between the client and the server was provided by the RMI Java technology. The desktop clients were implemented in a Java applet which allows its invocation over the internet by any browser. The mobile client uses the Java Micro Edition and the package Java ME RMI 1.0. The data generated by the prototype is stored on the server side in a MySql 5.0.21 relational data base. For the development of the agents community the Open Agents Architecture (OAA) was used, developed at SRI International. OAA is a framework for integrating a community of heterogeneous software agents in a distributed environment and is structured to minimise the effort of creating new agents. It allows the creation of agents in various programming languages and operating platforms and this particular detail will allows us to reuse agents executed in others works. With this kind of development tools we achieve an application platform independency.

In Fig. 10 it is possible to see the mobile version of the IGTAI. The mobile version for now is a small version of the IGTAI with limited functionalities.

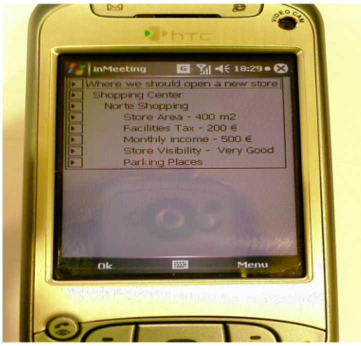

Fig. 10 Meeting mobile application

Fig. 11 shows the Meeting Desktop panel where it is possible to see the user ranking that consists of a three bar graphic representing the number of ideas introduced by the current user, the number of ideas from the worst and from the best one as well as the total number of ideas introduced. This will give a productivity idea and a little bit of competition which could lead to a productivity increase.

5 Conclusions

Nowadays, the human being expends a lot of time in working spaces, like offices and decision rooms. In spite of the amount of technology available (computer, networks, cameras, microphones), these spaces are still too passive. In order to make the working environments more smart or intelligent research is being done in the are of Smart Offices and Intelligent Decisions Rooms. this research originated some

Fig. 11 Meeting Desktop panel

prototype level spaces experiencing some of the functionalities that will be certainly available in the offices and meeting rooms of the future.

In accordance with the methodology proposed in [49] it is expected that Smart Offices and Intelligent Decision Rooms integrate AmI environments covering the following tasks:

- interpreting the environment state;
- representing the information and knowledge associated with the environment;
- modelling, simulating and representing entities in the environment;
- planing decisions or actions;
- learning about the environment and associated aspects;
- interacting with humans;
- acting on the environment.

The several systems presented in this chapter consider in part these tasks and gave the first steps for the future trends of this kind of environments.

6 Acknowledgements

We would like to acknowledge FCT (Portuguese Foundation for Science and Technology) and Programmes FEDER, POSI, POSC, and PTDC for their support in several R&D projects and fellowships.

References

[1] Analide C, Neves J (2002) Antropopatia em entidades virtuais. In: I Workshop de Teses e Dissertacoes em Inteligencia Artificial (WTDIA02), Brazil
[2] Andrade F, Neves J, Novais P, Machado J, Abelha A (2005) Collaborative Networks and Their Breeding Environments, Springer-Verlag, chap Legal Security and Credibility in Agent Based Virtual Enterprises, pp 501–512
[3] Asadi M, Huber M (2004) State space reduction for hierarchical reinforcement learning. In: FLAIRS Conference
[4] Brard F (1994) Vision par ordinateur pour la ralit augmente : Application au bureau numrique. PhD thesis, Université Joseph Fourier, June
[5] Brooks RA, Dang D, Bonet JD, Kramer J, Mellor J, Pook P, Stauffer C, Stein L, Torrance M, Wessler M (1997) The intelligent room project. In: Proceedings of the 2nd International Cognitive Technology Conference (CT'97), pp 271–278
[6] Chen F, Gao Y, Chen S, Ma Z (2007) Connect-based subgoal discovery for options in hierarchical reinforcement learning. 3th International Conference on Natural Computation (ICNC 2007) 4:698–702
[7] Cook DJ, Das SK (2007) How smart are our environments? an updated look at the state of the art. Pervasive Mobile Computing 3(2):53–73
[8] Davidsson P (2001) Multi agent based simulation: beyond social simulation. In: MABS 2000: Proceedings of the second international workshop on Multi-agent based simulation, Springer-Verlag, USA, pp 97–107
[9] Dodier R, Y DL, Riesz J, Mozer MC (1994) Comparison of neural net and conventional techniques for lighting control. applied mathematics and computer science. Applied Mathematics and Computer Science 4:447–462
[10] Fillinger A (Accessed in June 2008) The nist mark-iii microphone array. http://www.nist.gov/smartspace/mk3_presentation.html
[11] Fillinger A (Accessed in June 2008) The nist smart space project. http://www.nist.gov/smartspace/
[12] Frank R (2000) Understanding smart sensors. Measurement Science and Technology 11(12):1830, URL `http://stacks.iop.org/0957-0233/11/1830`
[13] Freitas C, Marreiros G, Ramos C (2007) IGTAI - An Idea Generation Tool for Ambient Intelligence. In: 3rd IET International Conference on Intelligent Environments, pp 391–397, Ulm, Germany
[14] Goel S, Huber M (2003) Subgoal discovery for hierarchical reinforcement learning using learned policies. In: FLAIRS Conference, pp 346–350
[15] Grudin J (2002) Group dynamics and ubiquitous computing. Commun ACM 45(12):74–78, DOI http://doi.acm.org/10.1145/585597.585618
[16] Heider T, Kirste T (2002) Intelligent environment lab. Computer Graphics Topics 15(2):8–10
[17] Huber GP (1982) Group decision support systems as aids in the use of structured group management techniques. In: Proc. of second international conference on decision support systems

[18] Huber GP (1984) Issues in the design of group decision support systems. MIS Quarterly
[19] Huber M (2005) Smart Environments: Technology, Protocols and Applications, Wiley, chap Automated Decision Making
[20] Johanson B, Fox A, Winograd T (2002) The interactive workspaces project: Experiences with ubiquitous computing rooms. IEEE Pervasive Computing 1(2):67–74
[21] Kang YB, Pisan Y (2006) A survey of major challenges and future directions for next generation pervasive computing. In: Computer and Information Sciences, ISCIS 2006, vol 4263/2006, pp 755–764
[22] Karacapilidis N, Papadias D (2001) Computer supported argumentation and collaborative decision making: the hermes system. Infor Syst 26(4):259–277
[23] Kraus S, Sycara K, Evenchik A (1998) Reaching agreements through argumentation: A logical model and implementation. Artificial Intelligence 104:1–69
[24] Kulkarni A (2002) Design Principles of a Reactive Behavioral S ystem for the Intelligent Room. Bitstream: The MIT Journal of EECS Student Research
[25] Le Gal C (2005) Smart Environments: Technology, Protocols and Applications, Wiley, chap Smart Offices
[26] Le Gal C, Martin J, Lux A, Crowley J (2001) Smartoffice: design of an intelligent environment. IEEE Intelligent Systems 16(4):60–66, DOI 10.1109/5254.941359
[27] Lewis L (1982) Facilitator: A microcomputer decision support systems for small groups. PhD thesis, University of Louisville
[28] Marreiros G, Sousa J, Ramos C (2004) Webmeeting - a group decision support system for multi-criteria decision problems. In: International Conference on Knowledge Engineering and Decision Support, Porto, Portugal, pp 63–70
[29] Marreiros G, Ramos C, Neves J (2005) Dealing with emotional factors in agent based ubiquitous group decision. LNCS 3823:41–50
[30] Marreiros G, Ramos C, Neves J (2006) Multi-agent approach to group decision making through persuasive argumentation. In: 6th International Conference on Argumentation ISSA 2006
[31] Marreiros G, Santos R, Ramos C, Neves J, Novais P, Machado J, Bulas-Cruz J (2007) Ambient intelligence in emotion based ubiquitous decision making. In: Artificial Intelligence Techniques for Ambient Intelligence, IJCAI 07, India
[32] Marreiros G, Santos R, Freitas C, Ramos C, Neves J, Bulas-Cruz J (2008) Laid - a smart decision room with ambient intelligence for group decision making and argumentation support considering emotional aspects. International Journal of Smart Home 2(2):77–94
[33] Marsa-Maestre I, de la Hoz E, Alarcos B, Velasco JR (2006) A hierarchical, agent-based approach to security in smart offices. In: Proceedings of the First International Conference on Ubiquitous Computing (ICUC-2006)
[34] Marsa-Maestre I, Lopez-Carmona MA, Velasco JR, Navarro A (2008) Mobile agents for service personalization in smart environments. Journal of Networks (JNW) 3(5):30–41

[35] Mikic I, Huang K, Trivedi M (2000) Activity monitoring and summarization for an intelligent meeting room. In: Workshop on Human Motion, pp 107–112
[36] Minder D, Marrón PJ, Lachenmann A, Rothermel K (2005) Experimental construction of a meeting model for smart office environments. In: Proceedings of the First Workshop on Real-World Wireless Sensor Networks (REALWSN 2005)
[37] Mitchell TM (1997) Machine Learning. McGraw-Hill Higher Education
[38] Moore D (2002) The idiap smart meeting room. IDIAP-COM 07, IDIAP
[39] Neves J (1984) A logic interpreter to handle time and negation in logic data bases. In: ACM 84: Proceedings of the 1984 annual conference of the ACM on The fifth generation challenge, ACM, New York, NY, USA, pp 50–54, DOI http://doi.acm.org/10.1145/800171.809603
[40] Nijholt A (2004) Meetings, gatherings, and events in smart environments. In: ACM SIGGRAPH international Conference on Virtual Reality Continuum and Its Applications in industry, pp 229–232
[41] Nunamaker JF, Dennis AR, Valacich JS, Vogel D, George JF (1991) Electronic meeting systems. Commun ACM 34(7):40–61, DOI http://doi.acm.org/10.1145/105783.105793
[42] Nunamaker JF, Briggs RO, Mittleman DD, Vogel DR, Balthazard PA (1996) Lessons from a dozen years of group support systems research: a discussion of lab and field findings. Journal Management Information Systems 13(3):163–207
[43] O'Keefe D (1990) Persuasion: Theory and Research. SAGE Publications
[44] Osborn AF (1963) Applied Imagination 3rd ed. Scribner, New York
[45] Petzold J, Bagci F, Trumler W, Ungerer T, Vintan L (2004) Global state context prediction techniques applied to a smart office building. In: Communication Networks and Distributed Systems Modeling and Simulation Conference, USA
[46] Petzold J, Bagci F, Trumler W, Ungere T (2005) Next location prediction within a smart office building. In: Gellersen HW, Want R, Schmidt A (eds) Third International Conference PERVASIVE 2005, Munich, Germany
[47] Prante T, Stenzel R, Röcker C, Streitz N, Magerkurth C (2004) Ambient agoras: Inforiver, siam, hello.wall. In: CHI '04: extended abstracts on Human factors in computing systems, ACM, NY, USA, pp 763–764
[48] Pruitt DG (1981) Negotiation Behavior. Academic Press, New York
[49] Ramos C, Augusto JC, Shapiro D (2008) Ambient intelligence - the next step for artificial intelligence. IEEE Intelligent Systems 23(2):15–18
[50] Russell SJ, Norvig P (1995) Artificial intelligence: a modern approach. Prentice-Hall, Inc., Upper Saddle River, NJ, USA
[51] Salancik G, Pfeffer J (1977) Who gets power and how they hold on to it: A strategic-contingency model of power. Organizational Dynamics 5:3–21
[52] Santos R, Marreiros G, Ramos C, Neves J, Bulas-Cruz J (2006) Multi-agent Approach for Ubiquitous Group Decision Support Involving Emotions, LNCS, vol 4159, Springer Berlin / Heidelberg, chap Ubiquitous Intelligence and Computing, pp 1174–1185

[53] Satyanarayanan M (2001) Pervasive computing: vision and challenges. Personal Communications, IEEE [see also IEEE Wireless Communications] 8(4):10–17
[54] Shrobe DHE (Accessed in June 2008) Intelligent room project. http://w5.cs.uni-sb.de/ butz/teaching/ie-ss03/papers/IntelligentRoom/
[55] Stanford V, Garofolo J, Galibert O, Michel M, Laprun C (2003) The (nist) smart space and meeting room projects: signals, acquisition annotation, and metrics. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003, vol 4, pp 6–10
[56] Streitz NA, Röcker C, Prante T, van Alphen D, Stenzel R, Magerkurth C (2005) Designing smart artifacts for smart environments. Computer 38(3):41–49
[57] Trumler W, Bagci F, Petzold J, Ungerer T (2003) Smart doorplate. Personal Ubiquitous Computing 7(3-4):221–226
[58] Waibel A, Schultz T, Bett M, Denecke M, Malkin R, Rogina I, Stiefelhagen R, Yang J (2003) Smart: the smart meeting room task at ISL. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, pp 281–286
[59] Want R, Hopper A, Falcao V, Gibbons J (1992) The active badge location system. ACM Transactions Information Systems 10(1):91–102
[60] Weiser M (1991) The computer for the twenty-first century. Scientific American 265(3):94–104
[61] Wellner P (1991) The digitaldesk calculator: tangible manipulation on a desk top display. In: UIST '91: Proceedings of the 4th annual ACM symposium on User interface software and technology, ACM, NY, USA, pp 27–33
[62] Yuizono T, Munemori J, Nagasawa Y (1998) Gungen: groupware for a new idea generation consistent support system. In: Proceedings 3rd Asia Pacific on Computer Human Interaction, pp 357–362
[63] Zachary W, Ryder J (1997) Handbook Of Human-Computer Interaction, Elsevier Science, chap Decision Support Systems: Integrating Decision Aiding and Decision Training, pp 1235–1258
[64] Zimmermann HJ (1999) Practical Applications of Fuzzy Technologies. Kluwer Academic Publishers, Norwell, MA, USA