ClassMATE: Enabling Ambient Intelligence in the Classroom

Asterios Leonidis[1], George Margetis[1], Margherita Antona[1] and Constantine Stephanidis[1,2]

Abstract—Ambient Intelligence (AmI) environments bring significant potential to exploit sophisticated computer technology in everyday life. In particular, the educational domain could be significantly enhanced through AmI, as personalized and adapted learning could be transformed from paper concepts and prototypes to real-life scenarios. In this paper, an integrated framework is presented, named ClassMATE, supporting ubiquitous computing and communication in a school classroom. The main objective of ClassMATE is to enable pervasive interaction and context aware education in the technologically augmented classroom of the future.

Keywords—Ambient intelligence, smart classroom, pervasive computing, education.

I. INTRODUCTION

THE concept of Ambient Intelligence (AmI) provides a vision of the Information Society where the emphasis is on greater user-friendliness, more efficient services support, user-empowerment, and support for human interactions. People are surrounded by intelligent intuitive interfaces that are embedded in all kinds of objects and an environment that is capable of recognizing and responding to the presence of different individuals in a seamless, unobtrusive and often invisible way [7]. AmI will have profound consequences on the type, content and functionality of the emerging products and services, as well as on the way people will interact with them, bringing about multiple new requirements for the development of the Information Society.

AmI is often claimed to bring a significant potential in the domain of education. Information and Communication technologies already permeate the classroom environment in many ways. They can play an important role in education by increasing students' access to information, enriching the learning environment, allowing students' active learning and collaboration and enhancing their motivation to learn [2].

However, the majority of the current research efforts in the field of AmI mainly focus on smart home or office collaborative environments, and few approaches address the need for an intelligent environment for students' education in the classroom itself.

For example, Interactive Classroom [4] promotes

[1] Foundation for Research and Technology – Hellas (FORTH), Institute of Computer Science, Heraklion, Crete, GR-70013, Greece. Email: {leonidis, gmarget, antona, cs}@ics.forth.gr
[2] University of Crete, Department of Computer Science, Greece

collaborative education, by proposing a virtual "whiteboard" shared via network among the students, through which they can exchange information or virtually attend lessons, without the need to be physically located in classroom's physical space. Such approaches, however, resemble the traditional web-based e-learning platforms where a "one size fits all" education is provided to learners [10], without taking into consideration either the individual learner's needs or the available environmental events to adapt and react properly. Such features, however, are considered to be fundamental from intelligent environments that attempt to augment educational procedure in the classroom.

On the other hand, the Smart Classroom is defined in [12] as a responsive intelligent environment that enhances collaborative learning using PDAs by introducing the concept of situation awareness, where a number of attributes are monitored (i.e., PDAs' locations, levels of noise and luminance, etc) to determine system response.

This paper proposes an integrated architecture for pervasive computing environments, named ClassMATE (Classroom Multiplatform Augmented Technology Environment), which aims to transform the conventional classroom into a context aware AmI environment. The next section presents a brief overview of fundamental issues concerning ubiquitous computing environments. Then, the ClassMATE architecture is presented, highlighting the key features that enable context aware education in the classroom, and finally conclusions for the presented work are discussed. ClassMate is currently being implemented in the context of the AmI Classroom Project of the ICS-FORTH AmI Programme [3].

II. RELATED WORK

According to ISTAG [7], a 'key enabling technology' for a successful implementation of the AmI landscape is the presence of middleware systems that act as the main coordinator of the heterogeneous and distributed services of an ambient intelligent environment. Nowadays, a plethora of such systems exist, providing the necessary functionality for the structured communication between the various AmI environments' components.

As mentioned in [11], the essential requirements that an AmI middleware should address are: heterogeneity integration, synchronous and asynchronous communication, resilience, security and ease of use. Moreover, a successful AmI middleware should provide all the necessary monitoring and control facilities of the diverse ubiquitous computing

artifacts which interoperate in an AmI environment, to make feasible a context aware orchestration strategy.

As an example, the AMIGO project ([9], [1]) developed a middleware that dynamically integrates heterogeneous systems to achieve interoperability between services and devices in intelligent home networks. The AMIGO architecture consists of a base middleware component, a programming and deployment framework and a series of legacy services named 'Intelligent User Services'. The base middleware component ensures the secure and robust interoperability of the heterogeneous service platforms that an AmI environment may host, providing also a generic mechanism for their semantic description of functional, non-functional and architectural features. The programming and deployment framework enables the developers to implement ad-hoc AMIGO-aware distributed services, allowing them to choose among two programming languages alternatives, .NET C# and Java. The Intelligent User Services, on the one hand constitute the legacy services layer of the AMIGO architecture that provides users with basic functionality to interact with the intelligent environment, while on the other hand they are responsible for compositing multiple information sources, disseminating any context-related information. Finally, any available information is encoded into a user profile and exploited towards environment's adaptation based on user's state and context changes.

The ClassMATE framework proposed in this paper addresses the needs for a robust and open ubiquitous computing framework in the school environment. It facilitates fundamental issues such as heterogeneous interoperability of AmI services, synchronous and asynchronous communication, resilience, security, context aware orchestration and ease of use in the intelligent classroom of the future. A key feature of the ClassMATE's architecture, that differentiates it from similar architectures, is the education-centric approach that has been adopted during its design. In terms of intecommunication, ClassMATE relies on a generic services interoperability platform, named FAMINE (FORTH's AMI Network Environment), which has been implemented in the context of the AmI Programme [3]. FAMINE provides the necessary functionality for the intercommunication and interoperability of heterogeneous services hosted in an AmI Environment. It encapsulates mechanisms for service discovery, event driven communication, remote procedure calls, etc., supporting a plethora of programming languages and frameworks, i.e., .NET languages family, Java, Active Script, ANCI C++, etc.

The next section provides a description of the CLASSMATE architecture, outlining research issues pertaining to context aware education in the classroom.

III. CLASSMATE

Intelligent classrooms in the near future will be populated with various devices, including interactive whiteboards, smart desks [8], students' personal mobile devices such as e-book

readers or tablets, etc. Figure 1(a) illustrates the orchestrating role of ClassMATE in an intelligent classroom, emphasizing its monitoring and coordination infrastructure.
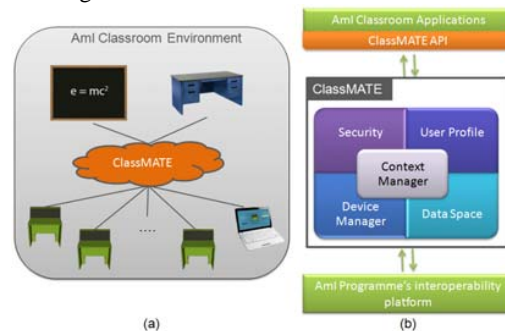

Fig. 1 a) ClassMATE provides a complete monitoring and coordination infrastructure for an AmI Classroom Environment. b) The ClassMATE architecture

In figure 1(b) the overall architecture of ClassMATE is depicted. Its major components are the ClassMATE's core and the API library. Finally, ClassMATE's backbone communicates with FAMINE, which is responsible for the overall service communication and data exchange. ClassMATE components are further explained in more detail in the next subsections.

A. ClassMATE Core

The ClassMATE's core consists of five major components: Security service, User Profile, Device Manager, Data Space and Context Manager. Each of these components is designed as a distributed service, which communicates with other services or applications through the FAMINE's infrastructure.

The Security service is responsible for the authorization management of the intelligent classroom's stakeholders (users and applications). It is based on a set of dynamically updated access lists, which define the rules that a user or an application must follow. The Security service is in continuous communication with the Context Manager, updating its access lists according to the current needs of the educational process. For instance, when the teacher requires the complete students' attention on the whiteboard, the Context Manager advises the Security service to suspend any interaction with other classroom's AmI artifacts (e.g., smart desks), thus students will not be distracted by the interaction with other devices.

The User Profile implements the classroom's users (students and teachers) behavior monitoring and evaluation, in order to provide user related metadata to the ClassMATE's services and applications. According to the IEEE's Learning Technology Systems Architecture (LTSA) [5], as illustrated in figure 2, the User Profile represents a learners' record repository, which keeps track of every individual student's learning status and behavior data. Additionally, the User Profile accommodates the knowledge learning resources library of students' behavior patterns, dynamically gathered via their activity monitoring. Data gathered by the User Profile service, through an iterative monitoring and evaluation procedure, constitutes the main feedback for the Context

Manager, so that a learner's centric rational is applied for content delivery and interaction control, thus providing adaptation to individual student's needs.
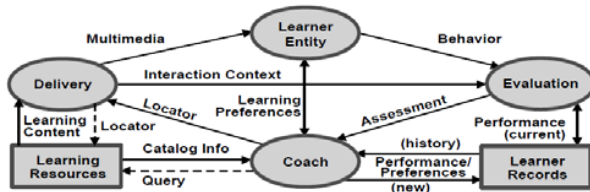

Fig. 2 IEEE's Learning Technology Systems Architecture

The Device Manager facilitates a generic mechanism for heterogeneous devices manipulation, both remote and local, by any ClassMATE enabled application. Every AmI artifact (interactive board, smart desk, etc.) that belongs to the classroom environment should accommodate a local Device Manager agent that handles input/output devices and support their interaction with any application in the ClassMATE cloud.

The Data Space's role is twofold: a) it implements a centralized content repository providing transparent content access and management by any ClassMATE application and service, as if it was a local resource, and b) it encapsulates a content classification mechanism, based on IEEE's LOM specification [6], providing the necessary content related rationale to data mining procedures. As figure 3 illustrates, the Data Space is able to gather material from diverse sources (e.g., web, file system, etc.) and correlate them into a content relational database by specifying their relationships based on a metadata scheme. The enrichment of content's metadata is accomplished offline by the teacher. That structured content can then be provided, through the context-aware mechanism to address different needs (e.g., application queries, students' course materials, lesson's courses, etc.). Consider the following example where the information provided by the Data Space about Pythagoras' work, the famous ancient Greek mathematician would differ for a student whose goal is an essay composition (e.g. historic references, images of him, etc.) than for a mathematical application, which tries to illustrate a list of Pythagora's axioms and theorems.
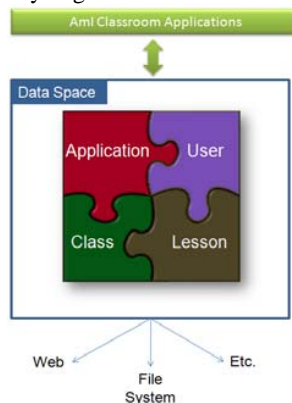

Fig. 3 The Data Space provides transparent access to all ClassMATE applications and supports content relational rationale

The Context Manager is the orchestration component of the ClassMATE's architecture. It is aware of any operation that takes place in the intelligent classroom (user interaction, sensor triggering, etc.). In more details, the Context Manager is responsible for making the decisions for every process workflow in the classroom's environment, and controlling the operation and collaboration of ClassMATE's services and applications to address users' needs at any specific time frame. To this purpose, the Context Manager applies appropriate reasoning strategies to user-, service-, application-related data, in the classroom environment.

Besides this general orchestration provided by the Context Manager, every AmI artifact in the classroom operates under the orchestration of a local Artifact Director, which is responsible for its robust operation. The Artifact Director, at any time, keeps track of what is currently running (applications or services) on the artifact, and according to the Context Manager's directions they initiate, stop or suspend the processes running on the artifact.

Figure 4 presents an example of how ClassMATE reacts to specific educational needs at a specific point in time. In this example, a teacher wishes to present the solution of an exercise on the interactive whiteboard. When the Context Manager recognizes that the teacher is going to present the related theorem on the whiteboard, it decides to suspend any other interaction with which students may be engaged in the classroom, in order to draw their attention on the whiteboard. To this purpose, the Context Manager directs the Security Service to forbid students' access to any AmI artifacts (e.g., their smart desks). The Security service updates its current access policy for students' access and propagates it to every device manager of the classroom. Concurrently, the Context Manager dictates all Smart desk Directors to forbid any new application launching on students' desk and suspend any already running applications. The Smart desk Directors suspend all currently running applications and request the Data Space service to store their current state in order to facilitate their future restoration. Finally, the Context Manager instructs the interactive whiteboard Director to launch the current course's exercises related application. While the teacher provides the exercise solution on the whiteboard, the Smart desk Directors initiate the personal assistant application on the Smart desk, which after consulting the User Profile, displays additional personalized information and examples as needed, according to the learner's profile.

B. ClassMATE API and applications

Complementary to the ClassMATE's core components, a software library is available for the development of ClassMATE enabled applications. The ClassMATE's library provides the necessary API for application – application and application – service intercommunication and interoperability.
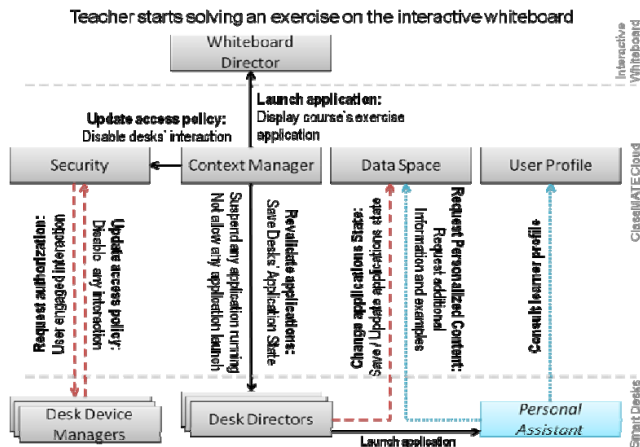
Fig. 4 Example of classroom's services and applications
orchestration by Context Manager

A key feature of ClassMATE's applications is their pervasiveness in the classroom, which is feasible by supporting the launch of every application in any classroom's AmI artifact. Figure 5 illustrates the aforementioned pervasive and collaborative application infrastructure. In more detail, a ClassMATE application consists of two major components: a thin client that encapsulates the application's UI and business logic, and a centralized state management service which is responsible for the synchronization of the different application's instances. Any application instance can be separately defined by the Context Manager as a passive application, thus only displaying the application and forbidding any interaction, or as an active application allowing its user to participate and collaborate to the interaction. For example, in the aforementioned scenario where the teacher solves an exercise on the whiteboard, the personalized assistive application launched on student's desk is a passive application, which only displays information.

IV. CONCLUSION

This paper has presented the ClassMATE architecture, a pervasive computing infrastructure for education, focusing on fundamental issues that should be covered in order for an AmI educational environment to be supported. The basic characteristic of such an infrastructure is its context awareness, in order to immediately respond and orchestrate the available AmI artifacts (e.g., interactive boards, smart desk, etc.) to address classroom stakeholders' needs effectively and efficiently. Furthermore, adaptive content retrieval mechanisms and a learners' behavior knowledge library should be established in order for the intelligent environment to provide the educational content appropriately adapted to each user's actual learning needs. Currently, a proof-of-concept version of these components was implemented and future plans include their complete implementation and exhaustive evaluation through a series of education applications for the smart classroom.
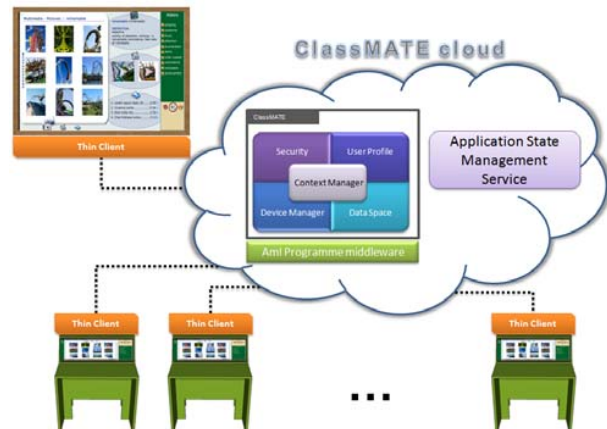

Fig. 5 ClassMATE pervasive computing and collaborative
application infrastructure

REFERENCES

[1] Amigo: Ambient intelligence for the networked home environment, http://www.hitech-projects.com/euprojects/amigo/index.htm, last visit at 04/06/2010
[2] Cook, D. J., Augusto, J. C., Jakkula, V. R. (2009). Ambient intelligence: Technologies, applications, and opportunities. Pervasive and Mobile Computing, Volume 5, Issue 4, August 2009, Pages 277-298.
[3] Grammenos, D., Zabulis, X., Argyros, A.A., Stephanidis, C. (2009). FORTH-ICS internal RTD Programme 'Ambient Intelligence and Smart Environments, In the Proceedings of the 3rd European Conference on Ambient Intelligence, Salzburg, Austria, November 18-21, 2009.
[4] H. Abut and Y. Öztürk, "Interactive Classroom for DSP / Communications Courses", in Proc. ICASSP 1997s, Munich, Germany, April 1997. Vol.: I, pp. 15-18.
[5] IEEE Learning Technology Standards Committee, "Draft Standard for Learning Technology-Learning Technology Systems Architecture (LTSA)", IEEE Computer Society, IEEE P1484.1/D9, 2001-11-30, 2001.
[6] IEEE LOM, "Draft Standard for Learning Object Metadata", IEEE Learning Technology Standards Committee, IEEE 1484.12.1-2002, 15 July 2002
[7] IST Advisory Group, Scenarios for Ambient Intelligence in 2010. Electronically available at: ftp://ftp.cordis.europa.eu/pub/ist/docs/ istagscenarios2010.pdf
[8] M. Antona, G. Margetis, S. Ntoa, A. Leonidis, M. Korozi, G. Paparoulis, C. Stephanidis, "Ambient Intelligence in the classroom: an augmented school desk", 3rd Interational Conference on Applied Human Factors and Ergonomics, to be published.
[9] M. Janse, P. Vink, N. Georgantas, "Amigo Architecture: Service Oriented Architecture for Intelligent Future In-Home Networks", Constructing Ambient Intelligence, pp. 371-378, Springer Berlin Heidelberg, 2008.
[10] P. Brusilovsky, "KnowledgeTree: a distributed architecture for adaptive e-learning" in Proc. 13th international World Wide Web Conference on Alternate Track Papers & Posters, New York, NY, USA, 2004.
[11] Y. Georgalis, D. Grammenos, C. Stephanidis, "Middleware for Ambient Intelligence Environments: Reviewing Requirements and Communication Technologies" in Proc. 13th International Conference on Human-Computer Interaction (HCI International 2009), San Diego, 2009, pp. 168-177.
[12] S. S. Yau, et al., "A Smart Classroom for Enhancing Collaborative Learning Using Pervasive Computing Technology", in Proc. 2nd ASEE Int'l Colloquium on Eng. Education (ASEE2003), 2003