Context – awareness and distributed events in mobile learning

Ivica Boticki and Vedran Mornar
Department of Applied Computing, Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia

Natasa Hoic-Bozic
Department of Informatics, University of Rijeka, Croatia

ABSTRACT

This chapter presents a system called MILE (Mobile and Interactive Learning Environment) which is used to support a blended approach to learning and teaching with mobile devices. The system has a modular and extensible system architecture which aims at supporting different platforms and devices both for students and for teachers. To better adapt to its users the system uses so called contextual widgets - components which gather, process and store contextual data. To disseminate education – related events, specifically designed Distributed Events Protocol (DEP) is used. Various applicative modules for mobile connected devices can be implemented upon the described architecture. They are, together with the experience gained in the project, described towards the end of the chapter.

Mobile learning, context-awareness, architecture, framework, collaborative learning, distributed events protocol, distributed information systems, e-learning, mobile computing

INTRODUCTION

Technology enhanced learning has always been of the great interest to educators. The capabilities of modern information and communication technology (ICT) have been perceived as a valuable force in dealing with mundane daily tasks such as exam correction or surveying. Even more intriguing has been the possibility of actually enhancing education so that students learn better (McCormack & Jones, 1997).

Still, computers can enhance educational process only if used correctly - in itself, the technology is neither good nor bad. ICT should be used so that it naturally adapts to its users. They shouldn't be overwhelmed with the capabilities a system offers and they should be able to completely focus on the learning and teaching activities. Therefore, a term named context – awareness becomes a buzzword in modern user – oriented systems.

The new technologies need to be complemented with modern teaching practices so today's e-learning forms are more oriented to communication, collaboration, and interactivity (European Commission, 2001) in both face-to-face (f2f) and virtual environments, overcoming the drawbacks of early versions of e-learning which used ICT primarily to improve learning content distribution.

Blended learning is becoming an increasingly popular form of e-learning, particularly suitable in the process of transition from traditional forms of learning and teaching towards e-learning (Alonso, López, Manrique & Viñes, 2005; Thorne, 2003). It combines traditional, or face-to-face teaching with ICT all wrapped in a pedagogically designed courses. Therefore, technology is seen as just one of many elements contributing to the successful teaching and learning.

Miniature mobile devices, ever-rising power and possibilities of communication (e.g. WiFi, 3G etc.) support an area of human activity that has long been technologically neglected: the life on the move. In addition to being used by individuals outside of their living and working environments, mobile technology is used by certain professions in order to acquire contextualized knowledge while on the move (Boticki, Mornar & Andric, 2006; Holzinger & Errath, 2007; Kukulska-Hulme & Traxler 2005).

In addition to desktop computers, mobile technologies begin to influence everyday activities and communication more and more and inevitably enter the world of education. The newest generation of mobile devices combines the power of yesterday's desktop computers with the unique characteristics of being personal, ambient and pervasive. Mobile technologies as a tool can support discursive elements of learning but should be built with the context – awareness in mind (Boticki, Mornar, Hoic-Bozic, 2006). A system designed in such a way can support education and give its social elements a completely new, mobile component.

The research on mobile learning at the Department of Applied Computing, University of Zagreb, Croatia was started in 2004 (Boticki, Mornar & Andric, 2006; Boticki, Mornar & Hoic-Bozic, 2006). After initial implementation dealing with specific, independent applications (e.g. mobile surveying), the need for a systematic approach to mobile learning environment was recognized. A multidisciplinary team had to be gathered in order to provide a pedagogical perspective on one hand and to include deep technological expertise on the other.

In order to support mobile learners in their contexts the system MILE (Mobile and Interactive Learning Environment) has been designed. The main purpose of this system is to support the didactic and the discursive approach to learning (Kukulska-Hulme & Traxler 2005). Students equipped with mobile devices of various kinds (laptops, tablets, PDAs and mobile phones), are connected to the system via faculty or campus wireless networks. They benefit from the central server component which distributes learning – related events to the tools available as a client side MILE application.

MILE's architecture is designed specifically by having mobile learners in mind. Users are not overwhelmed with technological tasks, such as the configuration of the system, filling in server addresses etc., since the system takes care of these in an intelligent way. Furthermore, a core component of the system, context engine, was built to take into account users’ context and adapt system’s behavior according to it. To achieve that, our context engine utilizes the concepts of subscriptions and contextually triggered actions.

Students using the system benefit from various tools, called modules, whose main aim is to enhance learning by enticing interaction and collaboration in a blended approach to learning. Some of the modules are MCollaboration, MNotebook and MWhiteboard and can be used in everyday classroom to support various educational activities.

The remainder of the chapter is organized as follows: first section of the chapter presents the background of the chapter including pedagogical issues, mobile learning and the use of contextual information in information systems and mobile learning and gives insight into the related work. The second section of the chapter presents the MILE system by describing its architecture. Special emphasis is given on the module for contextual information and the use of distributed events in the system. Towards the end of the chapter applicative modules of the MILE system are presented.

BACKGROUND AND RELATED WORK

Pedagogical Issues

Modern education is being transformed due to the emergence of the new technologies that strongly support constructivism in order to tailor the learning environments that best fit the needs of modern individuals making their learning as effective as possible. The constructivist school sees learning as an active process and emphasizes that „what is learned cannot be separated from how the learning occurs“ (Massey, Ramesh & Khatri, 2006). In that sense learning becomes more active, practical and problem based offering students more challenges when constructing their knowledge and extending their competences (Ally, 2004).

Information technology, together with the appropriate pedagogical design, has been found to facilitate the learning process in general. Therefore, learning management systems (LMSs) are gaining more and more popularity and are today used on a daily basis to support the education. However, a LMS tends to be more oriented on online learning with students passively reading contents prepared by their teachers. Therefore, the benefits of the classical, classroom – based, learning environment, often referred to as discursive elements, tend to be missing. Recent years have raised the awareness of that issue through the rise of blended learning (Alonso, López, Manrique & Viñes, 2005; Thorne, 2003) which combines classical methods of teaching with the ICT support.

Blended approach to learning uses various elements to motivate students in their everyday educational tasks. The blend of face – to face lectures, online learning, technology for content delivery and collaboration should help students in successfully completing the course (Bersin, 2004; Thorne, 2003).

The emphasis in blended learning is on the problem identification, blended model definition and careful monitoring of program implementation (Bersin, 2004). Thus a blended learning program should be well structured, with all steps well defined and announced to students in advance. It should use the most adequate technology available, save time both to instructors and students, create a social culture among the students and use experimental learning, peer group collaboration or problem based learning (Bound & Feletti, 2001).

Mobile learning

Since the beginning of the new millennium mobile devices have become an everyday asset to our lives. Their types range from simpler such as MP4 players to more complex communication enabled personal digital assistants (PDAs). Their technical capabilities are today quite high and open up many possibilities for their use in various everyday activities (Boticki, Mornar & Andric, 2006).

Since younger generations are especially passionate about mobile devices, their use in educational activities in inevitable. Mobile devices are perceived as unwanted by teachers which is not strange since their use in a learning environment is not controlled in an educational sense (Kaleidoscope Network of Excellence, 2006).

Although many debates on the appropriateness of use of mobile devices in education are being led, theorists agree on at least one thing: they strongly support "learning anywhere anytime" paradigm long promoted by the e-learning society. With the traditional technology learning has become independent of the time at which it takes place: one can learn at night or in the morning as long as he or she has access to the technology supported learning resources. Nevertheless, this classical approach heavily lacks the independence of users' location (Kukulsha-Hulme & Traxler, 2005).

The nature of mobile learning rose from the inherent capabilities of mobile devices as well as from the need of removing the location barrier when learning is concerned (Boticki, Mornar & Hoic-Bozic, 2006). Mobile devices have become ubiquitous and their communicational, computational, social and educational powers need to be used to enhance the learning in general. Mobile learning is ambient, that is aware of the surroundings at which the learning occurs, personal since mobile devices are our personal belongings and informal because it occurs whenever and wherever the need exists.

Collaborative activities are especially appropriate for mobility. Collaboration in technology supported learning has been limited by the static nature of desktop computers. Asynchronous collaborative activities such as forums support only certain kinds of collaborative activities, while synchronous communication depends too much on the location at which the hardware is installed. Since many modern mobile devices are communication enabled, they can transform social relationships into the medium for educational and didactical processes (Peters, 1999). Collaboration can be initiated at many levels, some of which are more formal (in-class or teacher – to – student) and the other more informal (on-campus or student – to student). Mobile learning therefore strongly supports the notion of learning as a collaborative activity.

Furthermore, mobility represents a force that inevitably leads to the transformation of education towards a more student-oriented, facilitator-oriented, informal and problem-based (Sharples, Taylor & Vavoula). The shift in the educational process is strongly supported by technology and reflects the competences required by the globalization. Project – based learning is seen as a key competence in today's business society: employees have to be team players, responsible, capable of grasping the bigger picture and be able to make decisions independently. The need for memorization of facts is no longer primary and is being replaced by the need for problem solving and lifelong learning. Individuals are required to manage on their own when dealing with daily tasks. This transformation is sometimes being referred to as to the loss of didactic substance (Peters, 1999).

From the technical point of view there are many projects dealing with mobile learning ranging from simple ones, utilizing mobile devices with preinstalled software to support all sorts of learning activities such as learning foreign languages via SMS (Markett, 2006) or reading e-books (Kukulska Hulme, 2007) to more complex custom made software usually aimed at supporting a specific learning activity such as museum tours (Hsi & Faith, 2005; Proctor and Burton, 2003), fieldwork (Pascoe, Ryan & Morse, 2000) or medicine (Holzinger & Errath, 2007; Kukulska-Hulme & Trayler, 2005). There are some quality reports summarizing projects and activities in the field of m-learning (De Freitas & Levene, 2003; Kukulska-Hulme & Trayler, 2005). Mobile learning community believes that current technological solutions shouldn’t be adjusted to fit the needs of mobile learners and that new technological solution (both hardware and software based) should be built from ground up (Kaleidoscope Network, 2006).

In addition to the hardware and software requirements, mobile learning systems can be evaluated by the pedagogical principles they use. They can be purely behavioral, cognitively oriented or designed to support situated approach to learning (Naismith, 2005). Although the relationship between the technical and pedagogical side cannot be firmly established, it seems that the support for the cognitive and situated approaches to mobile learning requires increased technical complexity when software is concerned (Kukulska-Hulme & Trayler, 2005).

Contextual information in information systems

The importance of usage of contextual information rises with the complexity of information systems. They should adapt to their users by anticipating their actions, communicating with other entities which are in some way connected with the user and by taking into account the state of user’s surroundings.

This way of adaptation to the state and user’s needs is called context-awareness and has been of a special interest to many researchers. This area of science is developing rapidly, a group of researchers has gathered experience through the number of accounts (Dey & Abowd, 2000; Dey, Abowd & Salber, 2001; Hazas, Scott & Krumm) and developed so called “Conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications” (Dey, Salber and Abowd, 2001).

The aim of the conceptual framework is to support the modeling of context-aware applications. It proposes models based on the main principles of software engineering with the most important being the ease of building, the support for reuse and the support for evolution. The authors of the conceptual framework translate these into the entities suitable for implementation into object oriented programming environments. Further in the chapter the framework will be referred to as “the conceptual framework”.

The classification of contextual information

Any information that can be used to characterize the situation of entities (whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves is called contextual information (Dey and Abowd, 2000). Following the definition of contextual information, the contextual framework suggest the classification of contextual information according to entities that can be found in systems and identifies four basic contextual information as entity characteristic: identity, location, status (activity) and time (Dey, Salber and Abowd, 2001). Identity maps unique identifier to an entity (e.g. unique academic person identifier mapped to students in Croatia), location describes spatial relations (e.g. room B1, ground floor, faculty FER, next to room B2), status is about intrinsic entity characteristics that can be inferred from the surroundings (e.g. temperature for room B1) while time is about better characterization of situations.

Basic context categories can be used to define or infer additional contextual information. Basic inferred contextual information are based on one contextual information (e.g. room number is derived from MAC address of a wireless network access point) while complex information get created from multiple contextual information describing the situation (e.g. a prearranged meeting cannot start until all required members are in the room and until a drawing board is turned on).

Context-aware services

Knowing, acquiring and deriving contextual information is the base for the so called context-aware services offered by applications:

- presenting information and services (e.g. when a student is positioned next to a room in which a lecture is held, a short lecture description depending on his or her interests is being displayed to him or her)
- automatically executing a service (e.g. when a student is in a room in which a teacher displays a presentation, the same presentation is forwarded to the student and the currently active slide is presented)
- attaching contextual information for later retrieval (e.g. student makes notes during the presentation which are then enhanced with contextual information to later determine to which course and slide they relate to) (Dey, Salber and Abowd, 2001).

Context-aware system characteristics

According to the conceptual framework, context-aware systems should be based on the following principles: separation of concerns, context interpretation, transparent and distributed communications, constant availability of context acquisition, context storage and resource discovery (Dey, Salber and Abowd, 2001). The separation of concerns is about separation of low level details (such as sensors gathering contextual information) from the higher level abstractions called widgets. Widgets hide implementation details, are reusable, support subscriptions and provide a well defined interface for communication with outer world. Context interpretations are about creating mutually connected application layers which deliver contextual information. Distributed and complex systems require transparent and distributed communication in order to hide network communication details and take into account global timeclock mechanisms. Since many software components use contextual information through the widgets and that a request for contextual information can arrive at any time, widgets have to provide constant availability of context acquisition. To reuse contextual information and predict trends contextual information has to be stored. Easy communication of all system entities is, amongst other factors, provided by a well designed mechanism of resource discovery.

Types of contextual components

In addition to widgets, the conceptual framework presents four additional categories of contextual components which can be used in designing a context-aware software solution: interpreters, aggregators, services and discoverers (Dey, Salber and Abowd, 2001). Interpreters raise the level of contextual information abstraction by typically using more sources of information (e.g. geo-coordinates can be seen as positions in a city or on a street). Aggregators aggregate related contextual information in a common repository and relate it to a concrete entity (e.g. aggregator for a group meeting can be activated only when all required meeting participants are in the same room). Services are components used to deliver actions depending on the information provided by widgets, interpreters or aggregators according to the application requirements (e.g. a presentation gets delivered to mobile connected devices only if an aggregator for group meeting gets activated). Discoverers take care of all system components (aggregators, interpreters and services) which can be used by applications. They provide a component repository and enable components' dynamic system inclusion.

Contextual information in mobile learning

In his work on context-aware mobile learning (CAML) (Yuan-Kai Wang, 2004), Wang identifies contextual information important for mobile learning. By building the theory of contextual mobile learning, he regards contextual information as contextual space dimensions and identifies the identity dimension, spatio-temporal dimension, facility dimension, activity dimension, student dimension and community dimension. Spatio-temporal and identity dimension map to the basic contextual information of the conceptual framework while the facility dimension could be seen as one of the basic contextual information since it describes the status of various technological components such as mobile devices, sensors, wireless network etc. Activity dimension describes higher order contextual information most often acquired by interpreters and aggregators such as class attendance, discussion participation etc. Student dimension describes intrinsic and psychological student characteristics such as the level of interest which are hard to determine automatically and is therefore, similarly to the activity dimension, based on the information acquired by interpreters and aggregators. The most complex is the community dimension which deals with the process of studying inside the community together with the status and interaction between members of community that make complex social contexts. Social context is about learning activities that evolve through time, place, educational institutions, home and practice making the role of student quite dynamic.

Similarly to the conceptual framework Wang (Yuan-Kai Wang, 2004) identifies four ways in which interaction in mobile contextual learning can be improved. Spatio-temporal dependent interface and contextual event notifications are related to the generic services that context-aware systems have to offer, while navigation and retrieval of learning materials and context-aware communication deals with the usage of contextual information in enhancing existing educational paradigms used in combination with mobile connected devices.

A generalized model for context-aware mobile learning system architecture

This section of the chapter describes a generalized model of context-aware mobile learning system architecture. Although it is just only one way of looking at the system presented in this section of the chapter, it clearly depicts the use of contextual information to support a learning environment based on mobile devices.

There are some examples of software using context-awareness, collaboration and virtual environments to support learning. Some of them discuss systems that facilitate educational information sources (Sakkopoulos, Lytras & A. Tsakalidis, 2006), others deal with mobile support for problem based learning (Massey, Ramesh & Khatri, 2006), discuss virtual learning environments with location-aware services (Brown, 1996; Morken & Divitini, 2005) or are focused on campus-wide initiatives (Griswold, Shanahan, Brown, Boyer, Ratto, Shapiro & Truong, 2004). They were used as a starting point when the development for the extensible, modular and platform independent architecture presented in this chapter was considered.

Following Wang's definition of CAML (context aware mobile learning) (Yuan-Kai Wang, 2004), the model possesses basic contextual dimensions called identity, spatio-temporal and the facility dimension. The basic contextual dimensions are used to generate the other, higher – order contextual dimensions. Identity is used to identify users; WiFi based location-awareness to provide system and users with location-enhanced services while spatio-temporal dimension and facility dimension are used to fine-tune system's technical characteristics such as devices' display orientation and size (i.e. display size can be adjusted according to the light intensity).

In order to support collaborative activities, the activity and community dimensions of context – aware mobile learning are utilized. Student activity is being aggregated, primarily to support collaborative activities. Community dimension is used to support pedagogical activities in which students form ad hoc learning groups. Community dimension of CAML in the MILE system can be used to support situated approach to learning (e.g. communities of practice).

For defining a model for contextual system architecture we should identify the contextual entities and the contextual information used by the entities. The model presented in this section of the chapter is designed to be universally applicable when designing system for teaching and learning with mobile connected devices.

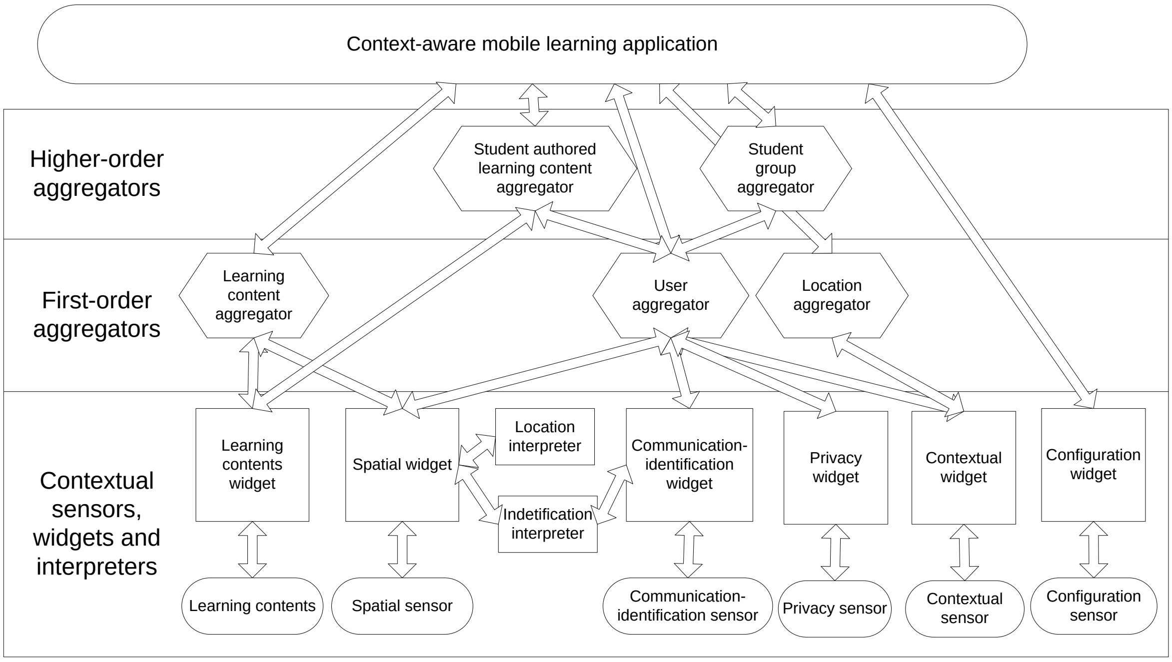


Figure 1. Three-tiered contextual system architecture for mobile learning

Figure 6 displays three-tiered contextual system architecture for learning with mobile connected devices. The lowest level is made of contextual sensors, widgets and interpreters. Contextual sensors acquire contextual information from physical or logical sensors. Research in the field of contextual information and contextual mobile learning (Dey & Abowd, 2000; Dey, Abowd & Salber, 2001; Dey, Salber and Abowd, 2001; Hazas, Scott & Krumm, 2004) identifies location as the most important contextual information. Communication-identification sensor is used to uniquely identify a user. Contextual sensor is used to acquire contextual information other than location, like the information on noise, light level, temperature etc. That information, according to researchers on context-awareness, can be used to adequately characterize the situation of entities and can be utilized in the realisation of a context-aware mobile learning architecture (Yuan-Kai Wang, 2004). In addition to the physical sensors, a logical learning content sensor is used to generate learning-oriented content to be used in the system (Dey, Salber and Abowd, 2001).

In addition to sensors, the lowest level of the generalized model of context-aware mobile learning architecture is composed of widgets encapsulating sensor functionality and enabling outer entities to create subscriptions to sensor events. The conceptual framework emphasises the importance of a widget object model in which existing sensors can be dynamically replaced to achieve better scalability and the natural use of discoverers. Location widget uses a location interpreter to interpret location sensor information as a specific spatial entity and an identification interpreter to translate user’s identification to user entity information.

First order aggregators communicate directly with the widgets. These entities use contextual information to adequately describe the situation of entities they present therefore enabling the number of different services. Students or teachers are the main actors of every learning environment and it is understandable to gather all contextual information related to them. Learning content aggregator contextually positions every learning material used in the system, while the location aggregator defines spatial context in which learning activities occur.

Higher-order aggregators are defined through the first-order aggregators. Examples include student authored materials (e.g. lecture notes) and learning group context determined according to the contexts of individual learners. In the realisation of the generalised model some contextual components can be placed on the server or on the client side, while other reside both on the server and client side uniting the possibilities of client and server modules.

MILE (MOBILE & INTERACTIVE LEARNING ENVIRONMENT)

System architecture

Main system characteristics

MILE is a context-aware system that supports learning and teaching facilitated by mobile connected devices and is designed to support everyday classroom activities and different pedagogical models. The system is to be used as a component in a blended approach to learning and teaching. In a typical scenario MILE is used to support just some learning and teaching activities, and as the blended approach to learning and teaching together with modern learning theories advise, is not intended to replace or to support all teaching and learning activities.

MILE is organized as a distributed learning network (Kaleidoscope Network, 2006; Kukulska-Hulme & Traxler, 2005) and supports distributed learning activities. Students use mobile connected devices, communicate and collaborate with teachers in order to fulfil their learning activities. Teachers use the system to deliver teaching activities as easy as possible without the negative interference of technology. System's server side coordinates students and teachers enabling various teaching and learning activities (Figure 1).

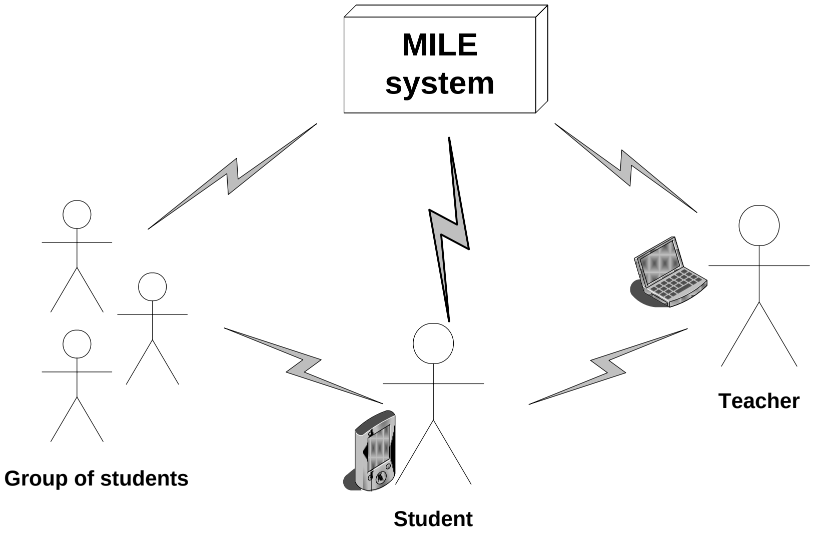


Figure 2. Main components and actors in the MILE system

The aim of the proposed architecture is to enable the use of mobile connected devices in a learning environment. When dealing with the use of mobile connected devices in the classroom difficulties can be experienced when adjusting current technological solutions to the actual needs of teaching and learning.

Following the advancements in the field of information technology, the architecture is designed to be modular, extensible, and service-oriented. It is based on object oriented principles, and is multi-layered and open.

Main system libraries

Main parts of the system are libraries called frameworks: the Base Framework, the Device Framework and the Server Framework. The latter two are built upon the Base Framework to provide services to specific parts of the system. The Desktop Framework is used by the applications for desktop computers; the Device Framework by the client application and its applicative modules while the Server Framework provides the base for the Contextual Information Service, Event Service, System and Applicative Services (Figure 2).

Applicative services are used by the applications installed on mobile connected devices to provide different functionality to MILE's users. Applicative services are commonly used as an interface to the central system data repository and to communicate with other server-side components. Applicative services are mutually independent and do not influence the operation of server services in any way which makes easily extendable and replaceable, even during the normal system operation.

In addition to services as basic building blocks, modular programming structures called libraries need to be created in order to provide common system functionality. The system is therefore divided into layers where higher level layers use the functionality of lower level layers through well defined interfaces (Figure 2).

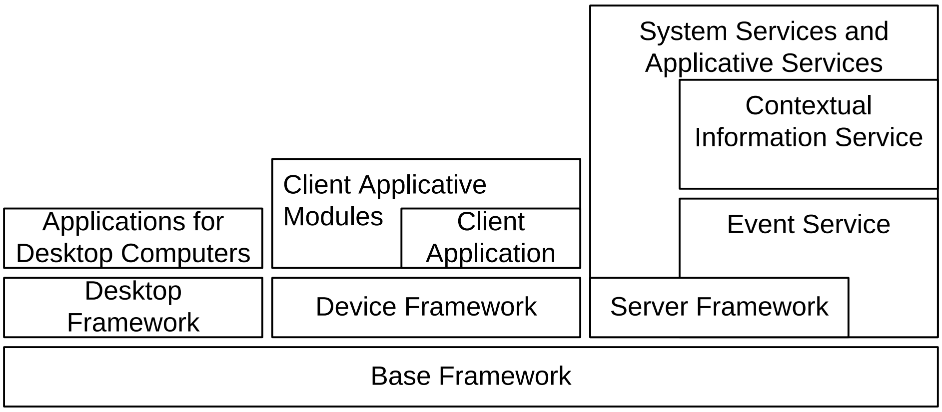


Figure 3. MILE system framework stack

Base Framework is a system component directly or indirectly used by other components. It is composed of sub-modules and classes designed to provide the means of communication through the Distributed Events Protocol (DEP): sending event messages from the server to the clients, structuring and assembling event messages. This library is composed of specially designed controls called widgets which are used to implement contextual features of the system: privacy, spatial, contextual, configuration and communication-identification widgets. All of them are used to exchange contextual information between mobile connected client devices and server components. Base Framework contains basic building components extended by the Device Framework and Desktop Framework in order to support device specific activity.

Server Framework is a component based on the Base Framework which provides services to higher-lever server components. Server Framework uses common modules for distributed events from the Base Framework and arranges the server side logic of the Distributed Events Protocol. The logic includes assembling event messages, sending event messages to clients, receiving information about the message delivery and retransmission of the message if the need occurs.

Server Framework uses configuration, location and contextual widgets. These widgets process contextual information on the server side before it is handed over to the module for contextual information to be stored in the database or to be further handed over to the module for event sending. The module for contextual information is used to receive contextual subscription (explained in detail in the chapter "Contextual subscriptions and actions in the MILE system") and contextual requests for event message sending (Figure 3).

In addition to using widgets to deliver contextual system architecture, server framework contains modules for data caching in order to ensure fast data access as well as numerous data and parameters specific to the server side of the system.

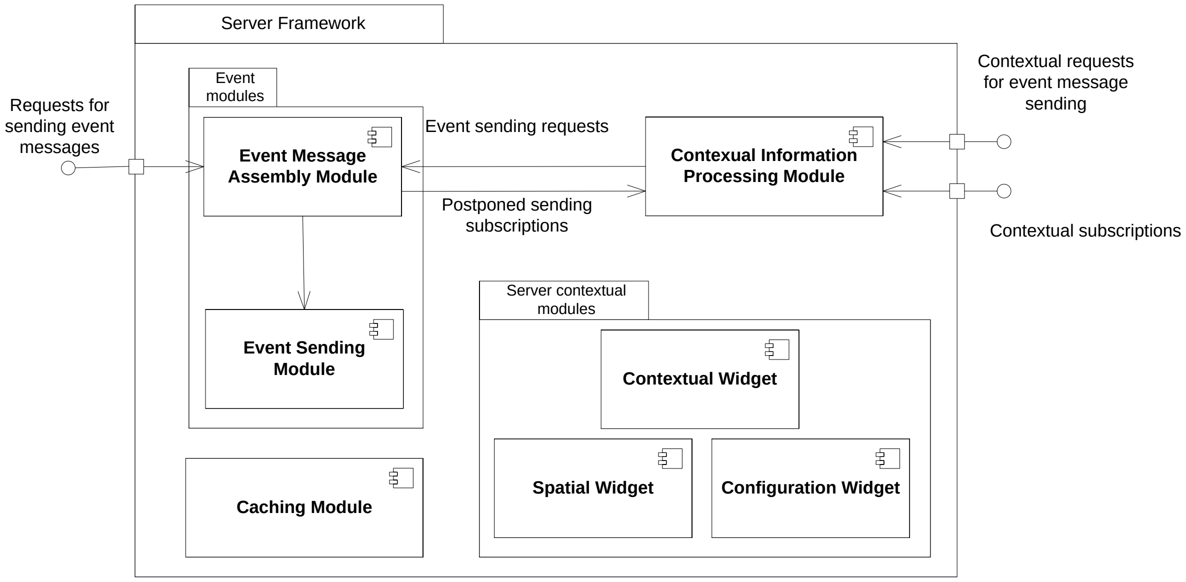


Figure 4. The Server Framework

Device Framework uses widgets to enable contextual system architecture by extending base widgets' functionality available within the Base Framework. By extending the Base Framework components, new components that enable information exchange between contextual client (mobile connected device) and server components are made possible (Figure 4.).

Device Framework contains the elements for the client side of the Distributed Events Protocol (DEP) (explained in detail in chapter “The use of distributed events in the MILE system”), most important of them being the module for message receival and response. To represent an event message with appropriate programming entities, a special module called the Event Factory connected with the mechanism of the Distributed Events Protocol (DEP) is used.

Since all applicative functionality of mobile connected devices in the MILE system is realized as a set of loosely connected modules which can easily be plugged in and out from the system, this library contains additional modules used to drive the communication. To provide secured access to services, a mechanism for acquiring service tickets from the server side and storing them in a special component called Ticket Store is provided. Stored service tickets can be (re)used by all client applicative modules.

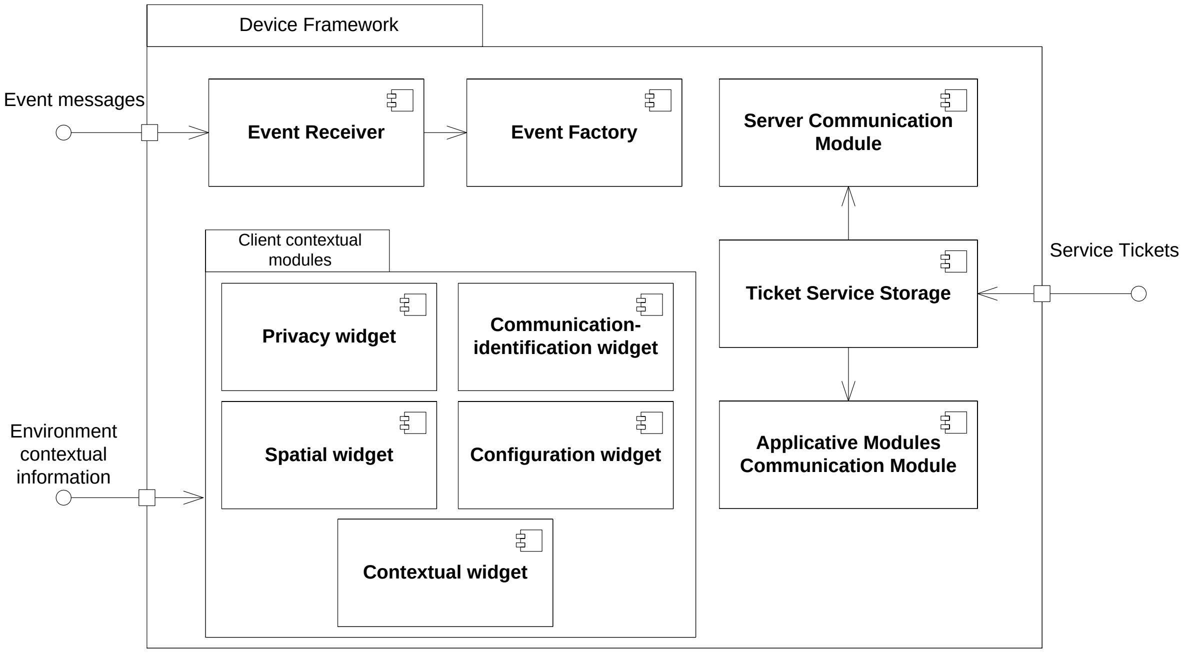


Figure 5. The Device Framework

Desktop Framework is used to build applications for desktop computers used primarily by teachers in order to track and coordinate events in a learning environment supported by mobile devices. Since desktop computers differ from mobile connected devices by numerous hardware and software characteristics (network connectivity, display size etc.), it was necessary to design the Desktop Framework as a special component. The most important elements of this component are specific implementation of contextual widgets: privacy, spatial, contextual and configuration-identification widgets with their functionality derived from the Base Framework. Furthermore, proxy modules used to access server side services are contained within the framework taking into account the limitations of the network capabilities of the mobile devices.

Privacy widget is used by the end users to enter or to leave the private mode of use while the configuration widget disseminates the configuration data to the mobile clients. Spatial widget is separated from the contextual widget due to the specific nature of spatial contextual information.

Main system services

MILE has a complex architecture designed to support different learning and teaching activities. It coordinates teaching activities provided by teachers and learning activities in which students participate. Within its service – orientated architecture several groups of service can be identified: core services, system services and applicative services (Figure 2).

System services are primarily consumed by client applications. Clients receive spatial information by using the Spatial Service, report contextual information by using the Contextual Service, perform automatic configuration using the Configuration Service and use Communication-identification Service to be registered as system users and therefore receive access to other services provided by the system. Authentication Service is used to log into the system, Registration Service to register system users, Security Service to ensure data security, Ticket service to be used by other services as a supplement to the Security Service and Subscription Service to provide access to and manipulation of contextual subscriptions.

Core services are the main components of the system and are used to deliver events to all subscribed system users. Events are generated with the help of all other services or by the Contextual

Service. Contextual Service generates events based on contextual subscriptions and current users' context data and hands it over to the Event Service. Event service can be directly used by all server side components. This module encapsulates functionalities for event sending and separates the details of transformation of an event message from a programming entity into a network transferable entity (Figure 3). Event service is used by the Service for contextual information which, depending on the situations in a learning environment or created subscriptions, creates events to be delivered to all interested parties in the system. Contextual Service analyses data stored in the contextual database scheme in order to produce a list of events to be sent to clients. The analysis is preformed depending on the user request, subscription or automatically on a periodical base.

Applicative services are built together with the applicative modules as needed by course designers. Applicative services can utilize core and system services in order to deliver application-specific functionality to mobile devices.

The client application and applicative modules

Applications for desktop computers use available system architecture through the Desktop Framework (Figure 2) to provide teachers with the possibility for delivery and coordination of educational activities of chosen pedagogical models. Teachers use applications to see all students present in the learning environment, to share educational materials with students, to entice students in their interaction and collaboration or to engage in the interaction and collaboration with students supported by the MILE system.

The Client Application is available on the students' connected mobile devices and is primarily used to support student educational activities. The Client Application receives education related events through the system infrastructure and forwards them to client application modules to be utilized for educational purposes (Figure 4). The Client Application is a host application which can fetch, run or turn off applicative modules through the designated services. The advantage of this approach is that in the case of change in the applicative modules, client application doesn't have to be modified or upgraded. The Client Application provides user with numerous technical details about the system such as contextual information on privacy, locations, network status etc.

Client applicative modules are used to provide specific applicative functionality to clients involved in educational activities of used pedagogical models.

Applicative services are independent components built on top of the Server Framework, Contextual Information Service and Event Service. For the operation they use database as data storage, custom applicative logic and other system services. They use system services as an interface towards the commonly used system level operations and use the Event Service for sending different events to different system users (Figure 3).

The communication model

The system communication model is based on a client-server communication model and uses TCP/IP network protocol suite (RFC 675, 1974). There are two basic communication areas: client-server communication area and client-client communication area. Client-server communication area is the base of MILE's system architecture and enables the communication over the HTTP, TCP and UDP protocols as well as over the specifically designed Distributed Events Protocol (DEP) which is discussed in detail in the section "The use of distributed events in the MILE system". Client-client communication area, although controlled by the Device Framework, enables various network activities based on TCP and UDP protocols (Figure 5).

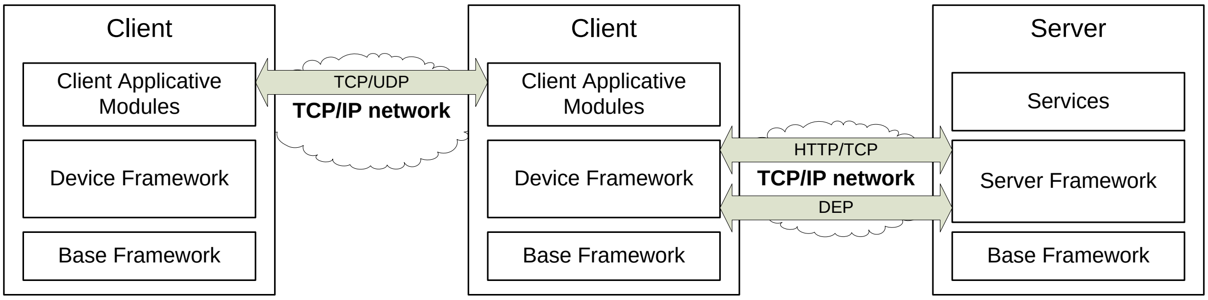


Figure 6. The communication model

The client-server communication area is used to access system and applicative services and to exchange event messages. System and applicative services are accessed by HTTP or directly by TCP protocol depending on the system configuration. The Device and Server Framework provide communication services to higher level layers: client applicative modules and server applicative and system services. The communication with the services is always initiated by the clients while the event message sending is reserved for the server system side of the client-server communication area. In the latter case the communication is always initiated by the server and is based on the Distributed Events Protocol (DEP).

Client-client communication area is used for communication between clients and is realized at the client application applicative modules level. In contrast to communication with the server, the communication between client modules is not controlled by the server and is designed depending on the applicative requirements (e.g. chat, binary file transfer etc.).

Three-tiered data architecture

Like many business applications, MILE system if composed of three basic layers: presentation layer, business logic layer and data access layer. Data access layer, considered to be the lowest layer, is used to store, fetch and manipulate data. It should have good technical characteristics because it is intensively used by all parts of the system. The system database is considered to be a part of the data access layer as well. The Business logic layer encapsulates system business rules and is used to model business processes that work with business entities. The Presentation layer is applied on top of the business logic layer and delivers the states of business entities to users.

The system is composed of system and applicative services that require data storage. The data architecture in MILE is designed according to the up-to-date paradigms of software engineering and practices (Microsoft patterns & practices Developer Center). It contains a main database component. Furthermore, data access is organized in layers with the main elements of the system represented as business entities and processes. This way of system development enables efficient development, error identification, upgrade and maintenance of the system.

The module of contextual information

Contextual modules in the MILE system

The implementation of the MILE system is based on the generalized model of context-aware mobile learning architecture. During the realisation several specializations of the model had to be done according to some physical limitations. Nevertheless, major conceptual changes to the elements contained in the contextual model have been avoided.

Client side sensors include communication-identification and privacy sensors. Distributed sensors are spatial, contextual and configuration sensor. Learning material sensor is a distributed logical sensor primarily used to fetch learning contents to the user.

Contextual widgets in the MILE system

Client contextual components of the MILE system collaborate in order to deliver system functionality. Mobile connected devices use contextual information from the learning environment to get connected to a network, fetch the data from the server and communicate with the server (Figure 7).

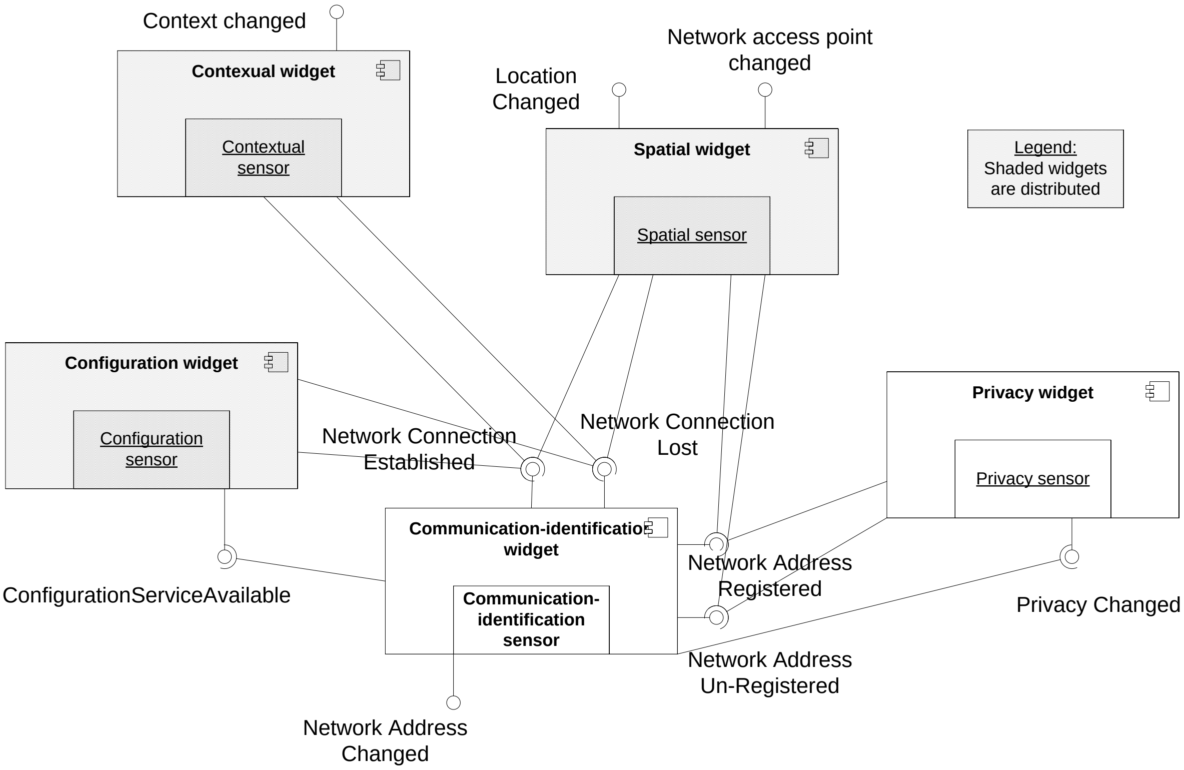


Figure 7. MILE’s contextual widgets

Communication-identification widget represents a central component of the contextual client in the MILE system and provides some basic communication-identification services to users. The widget is made of two timers used to gather information on network connectivity and network address. Wireless network connectivity is versatile contextual information because of frequent position change and low signal coverage. Therefore it is necessary to test network connectivity periodically and report it to the client application. In addition to changeable network connectivity, mobile connected devices tend to change their network addresses (e.g. IP addresses). Although the situation in which mobile client is assigned a new network address without the loss of connectivity is quite rare, client network identification is often used in the system and it is necessary to ensure it is precise and up-to-date. To achieve that, another timer periodically gathers the information about the network address. Communication-identification widget reacts to the changes of mobile connected device network address and privacy widget's settings. It reports mobile device address change to the server-side components.

Configuration widget is used to provide information about the system server side. Mobile clients have to fetch information about the service access points (addresses) and the ways of communication with services. Configuration widgets look for server on all available network interfaces by broadcasting a specially prepared UDP package. After receiving a client request, configuration server sends to the client detailed information about the way of communication with the server. After receiving the information, communication-identification widget establishes the communication with the server (e.g. registers its IP address).

Privacy widget provides contextual information about the privacy defined by the user. Privacy is considered to be a very important characteristic of context-aware systems since they reveal some information that can, under certain circumstances, be regarded as classified or too personal to be shared (e.g. teacher's location when in a meeting). Therefore, the privacy widget transfers user requests to the communication-identification widget which then performs the registration or the un-registration of users by using communication-identification service. The feedback on the outcome of user registration attempt is submitted back to the privacy widget.

Spatial widget is used to gather contextual information from mobile clients, interpret it by using server side spatial component and to deliver all required information to other client widgets. This widget depends severely on network availability and information provided by the communication-identification widget. Only after the communication network is available, the automatically assigned MAC network address (IEEE Std 802, 2001) can be identified. If the network address is registered, widget communicates with the server side spatial module to interpret the information about the access point and retrieves a learning environment location identifier. It can be used on the client and on the server side to provide numerous context-aware services.

Contextual subscriptions and actions in the MILE system

One of the most important elements of context-aware systems is contextual subscriptions. Contextual subscriptions are contextual entities which are triggered by certain contextual prerequisites within the specified time interval. Contextual subscriptions can be defined by the services or by the Event Sending module (Figure 3). Services define subscriptions in order to implement system or application logic while the Event module uses them to store undelivered event messages. Each subscription consists of so called actions which define what should be done when the subscription gets triggered.

There are four possibilities for contextual subscription:

- Contextual user can be subscribed to a context of a user
- Contextual user can be subscribed to a context of a group
- Contextual group can be subscribed to a context of a user
- Contextual group can be subscribed to a context of a group

As an example of the last subscription type, an educational situation in which inter-group collaboration is enticed and in which a message is sent to some groups about a contextual change in another (e.g. members' location change). Action triggered in that case is called "LocationChanged" and is aimed at all subscribed members or groups.

To support the system of contextual subscriptions and actions, active contextual actions and their receivers can be extracted in three ways:

- Periodical data storage analysis for all users and subscriptions
- Data storage analysis for a specific user

- Data storage analysis for a specific subscription

Described components are realized as database stored procedures in the MILE data storage and optimized according to the best practice in order to provide the most efficient results.

The use of distributed events in the MILE system

During the activities of learning and teaching with mobile connected devices a need for transferring information about numerous events between main system actors occurs. Just by being active, clients explicitly spawn many events which are then transferred to all interested system entities. If a teacher opens a presentation on his mobile connected device, students' client applications are notified of the event and react to it appropriately (e.g. fetch presentation opened by the teacher from the central server repository).

Although most events get generated explicitly by students and teachers, some are generated implicitly as a result of server services operations and existing contextual subscription. Events are generated implicitly on the server side when subscriptions get active and are being sent to all subscribed users (e.g. student as a subscription author might have requested to be notified when his colleagues for informal learning enter the room D1). When the context of the subscription is met the server initiates communication and sends event information to all interested users registered as active on the network (Figure 8).

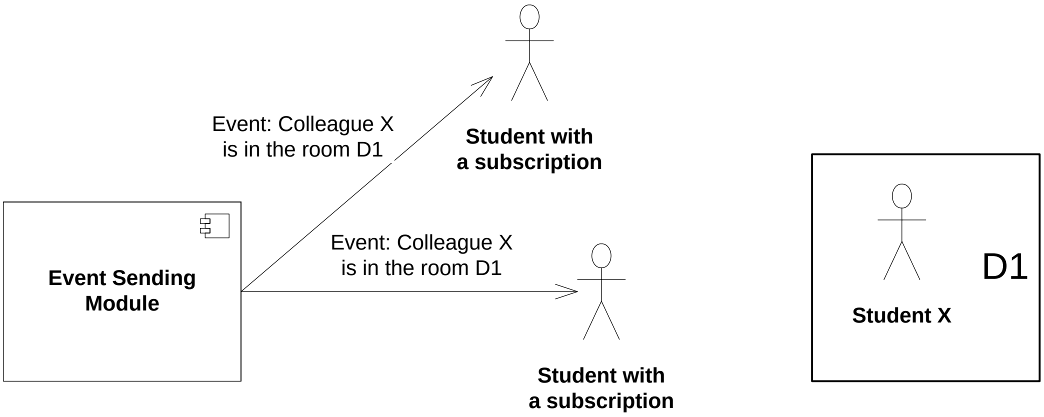


Figure 8. An example of contextual subscription realization

Events need to be transferred in relatively short time to all users they are aimed at. Since there are typically many users in the system and the number of events can get quite high, a special care has to be taken in order to make balance between the use of network resources, response time and reliable event delivery. As an appropriate and tested in practice solution, a TCP/IP protocol family was chosen and its UDP transport layer protocol for package delivery.

In order to ensure the independence of the network transportation mechanism on the one hand, and to satisfy the requests for low network traffic, adequate transfer speed and reliability on the other hand, the Distributed Events Protocol (DEP) was designed. It is used to exchange so called event messages (or DEP messages) which transfer event information and various technical-communicational data as the receiver's and sender's network addresses.

UDP as base protocol for the Distributed Events Protocol

Although not as popular as TCP, UDP (eng. User Datagram Protocol) is often applied in specialized systems like real time systems and internet telephony (VoIP – voice over IP) (Jackson, 2007). The UDP protocol is used to exchange short messages called datagrams. It does not guarantee the package delivery or ordered packages. Packages therefore can get lost, be delivered more times or be wrongly ordered. On the other hand, UDP has much shorter diagrams and in practice much faster delivery of packages. It is seen as a protocol for time-sensitive applications and is used in systems with many clients and small amounts of data to be transferred between clients and servers. Furthermore, the advantage of UDP over TCP is in the use of broadcast or multicast mechanisms to deliver a single datagram to more addresses which minimizes network load and utilised physical network infrastructure.

UDP is due to its all characteristics appropriate to send events in the MILE system:

- Events are relatively small chunks of data and can be packed as datagrams which in theory can deliver 65,535 bytes (in practice that number is lower and on IPv4 is 65.507 bytes)
- Events need to be transferred as fast as possible. Real time transfer enables high quality service on the client side
- Events can be seen as isolated entities and don't have to be delivered in a specific order other than defined on the application level
- Event transfer should require low system and network resources. UDP enables these since it is not ordered, there are no sessions etc.

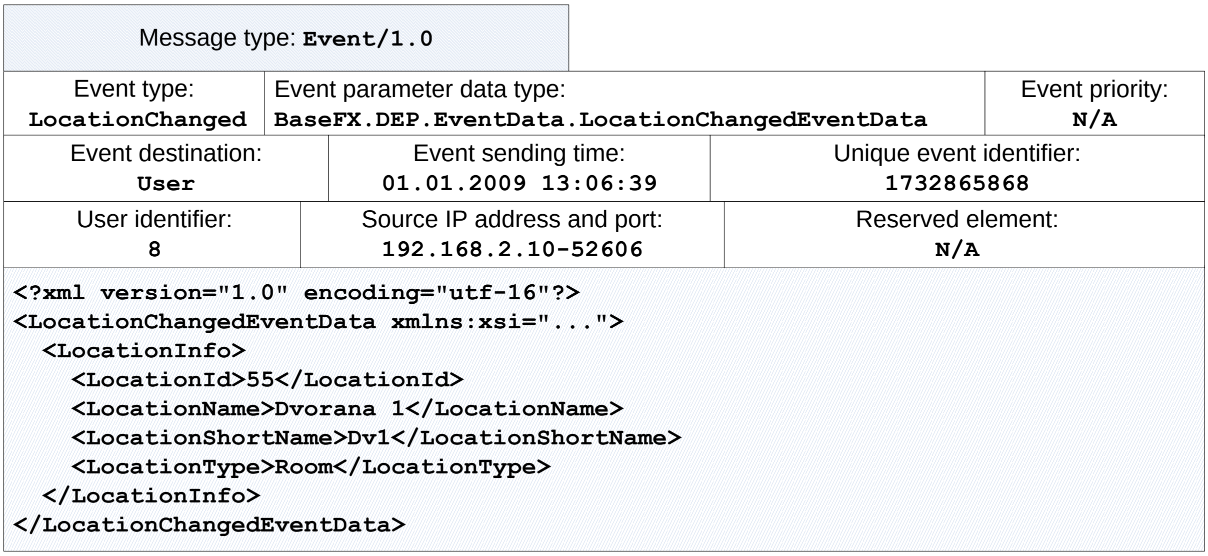
Nevertheless, UDP as a candidate for the distributed events protocol has one serious drawback: there are no mechanisms for the received message (UDP datagram) confirmation. Therefore, the Distributed Events Protocol (DEP) should, among other things, ensure event message delivery.

Event messages

DEP transfers specially designed event messages from senders to receivers. Since every sent event message has to be confirmed with an event response message there are two types of messages: event messages and event response messages.

The logical view on the event message describes it as a system entity, while the physical describes concrete data transferred from the point of sending to the point of receival. On the physical level an event message is composed of message headers and of a message body. Event type (EventType) defines an event in the MILE system, event parameter data type (EventDataType) is a fully qualified data type name describing an event parameter. This header is reserved for simple information about the parameter type being sent, while the data itself can be found in the event body (e.g. for a location change event, event parameters will be the data about the user who changed his or location). Event priority (EventPriority) is information which is transferred to the transport layer in order to determine the priority of the message to be sent. Event destination (EventDestination) contains information on the receiver's type (individual user or group of users), sending time (EventTime) contains the time when the message was sent, unique event identifier (EventId) is temporary attached to an entity to uniquely identify events in the system when sending and responding to event messages. User identifier (UserId) describes a user at the system level to whom the event message is sent. Source IP address and port (SourceIPAddress) contains the information about the sender to whom the event response is to be sent. Reserved element (RFU) is used for future purposes.

An example event message structure used to send information about the user's location change event is depicted in Figure 9.



Message type: **`Event/1.0`**

Event type: **`LocationChanged`**	Event parameter data type: **`BaseFX.DEP.EventData.LocationChangedEventData`**	Event priority: **`N/A`**
Event destination: **`User`**	Event sending time: **`01.01.2009 13:06:39`**	Unique event identifier: **`1732865868`**
User identifier: **`8`**	Source IP address and port: **`192.168.2.10-52606`**	Reserved element: **`N/A`**

```
<?xml version="1.0" encoding="utf-16"?>
<LocationChangedEventData xmlns:xsi="...">
  <LocationInfo>
    <LocationId>55</LocationId>
    <LocationName>Dvorana 1</LocationName>
    <LocationShortName>Dv1</LocationShortName>
    <LocationType>Room</LocationType>
  </LocationInfo>
</LocationChangedEventData>
```

Figure 9. An example event message for the location change event

The Distributed Events Protocol

DEP is an applicative TCP/IP layer protocol which uses Internet layer UDP protocol. The need of designing a special protocol together with its event and event response messages came because of the specifics of the MILE system, the most important being fast event transfer and low network traffic. The new protocol is justified by the comparison of the round trip time for message delivery over TCP and DEP. For single message delivery the measured time is of the same order of magnitude (slightly in favor of DEP), while for multiple messages DEP ensures much faster delivery.

In a typical scenario of an event message sending procedure, the server initiates communication by sending message while clients respond to it with the event response message (Figure 10).

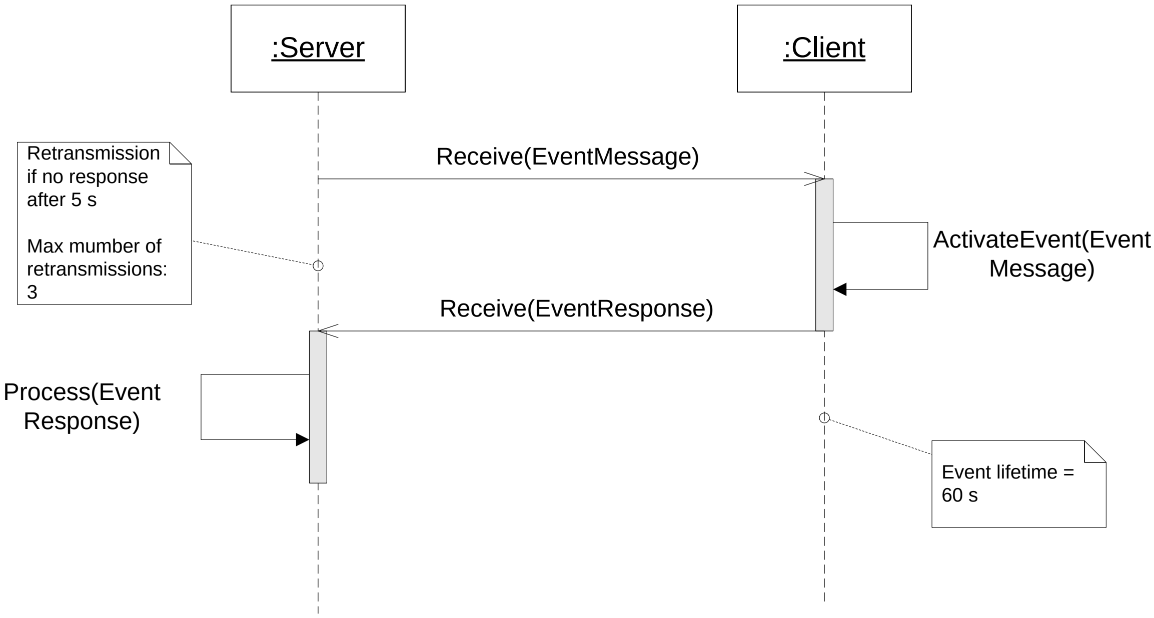


Figure 10. Event sending and responding to an event in the Distributed Events Protocol

For dealing with situations where the server sends more than one event message in parallel, a common port range is used. Before sending event message, the server fetches two available port identifiers from the common range (e.g. [50000, 55000]) and uses one of them to send the event message and the other to receive event responses. All clients, on the other hand, use a reserved port (e.g. 12345) to receive event messages and to send event response messages. If the server does not receive a message response the process of message sending is repeated for a specified number of times (e.g. 3). Should the answer be lost again, the event sending procedure is aborted.

The distributed events protocol is a distributed algorithm realized on the server and clients and uses certain synchronization mechanisms. The complexity of the protocol is in the parallelism that needs to be introduced. On the server side parallel message sending and receiving with the mandatory synchronization points is implemented. Immediately after receiving an event message client responds to it with an event response message. Client tracks unique event identifiers and eliminates duplicate events.

A special care has to be taken of the history of received event messages. The server might resend an event message if a response message is not received in time. This is possible even when a client has already received the event message and its response hasn’t reached the server. To solve this problem the server tracks down event message history and discards duplicate event messages.

APPLICATION MODULES

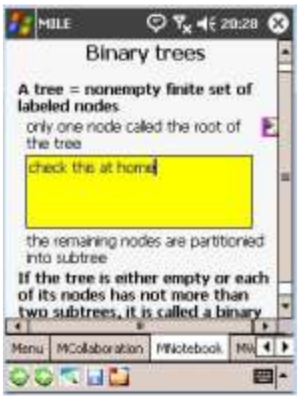
MILE offers many different modules that support a learning environment based on mobile devices, some of the most interesting being:

- MCollaboration - used to support collaboration
- MWhiteBoard – an alternative to a classical blackboard available to mobile students and teachers
- MNotebook – used to deliver PowerPoint presentations to mobile devices
- MSchedule – customizable scheduler for mobile devices
- MSurvey – instant in-class mobile assessment and surveying
- MAccessibility – accessibility features for users with special needs
- MVirtualBoard - notice board available anytime anywhere
- MGuide - guide to current education related events

This chapter examines in detail modules that are the most important when collaborative approach to learning and blended teaching model are concerned. According to the constructivism, learning should be interactive to promote higher-level learning and social presence (Kaleidoscope Network, 2006). By using MILE modules students interact online with other students, instructors and the content.

MNoteBook module

Listening to a lecture, following a displayed presentation, making notes and exchanging them with colleagues all at the same time is challenging even for the most skilful students. MNotebook module is used to automatically display presentations opened by a teacher in a classroom. As the teacher passes on to the next slide, slides are automatically changed on students' mobile devices. In addition to that, students are given the opportunity to tag the bullets with their own notes (Figure 11).



Binary trees

A tree = nonempty finite set of labeled nodes

only one node called the root of the tree

check this at home

the remaining nodes are partitioned into subtree

If the tree is either empty or each of its nodes has not more than two subtrees, it is called a binary

Figure 11. An MNoteBook slide shown to a mobile user

After being created, notes are uploaded to a community web site and become available to other students as a secondary learning resource with the information of significance at a student community level, providing students the possibility to benefit from other students' notes thus facilitating the creation of social networks and clearly demonstrating socializing powers of mobility.

The module presents a process some authors refer to as the digitalization of didactic action since it transforms elements of classical education and inevitably leads to a shift of competences when some standard student activities are concerned (Peters, 1999). In addition to that, such an approach aids recall and reflection similarly to other form of social software such as blogs or discussion groups (Kukulska-Hulme & Trayler, 2005).

MCollaboration module

The module supports collaborative activities such as chat, file sharing and instant messaging by utilizing wireless network and MILE's central server – side component (left hand side of Figure 12.). Students are notified of the presence of their colleague in the learning environment and are able to, depending on the security and privacy settings, explore colleagues' whereabouts through a digital map of a learning environment (right hand side of Figure 12.).

Figure 12. Capabilities of the MCollaboration module

Socializing powers of mobility are said to facilitate collaborative learning and encourage interaction (Hoic-Bozic, Boticki & Mornar, 2005). Learning is inherently social and mediated through interaction with others, and involves a process of engagement in a community of practice. The MCollaboration module strongly supports the concept of communities of practice, with members sharing common interests, learning together and interacting with each other (Mayes & De Freitas, 2004). By using MILE communities of practice can be formed spontaneously and can be very informal, or they can be formed „artificially“ (e.g. by a teacher). Either way the important role of social interaction and collective learning is strongly emphasized.

MWhiteBoard module

MWhiteBoard is a module that supports group work in a controlled classroom environment. Students collaborate by sketching ideas on their mobile devices and sharing them on a common canvas (e.g. on a projector) (Figure 13.). In this way knowledge can easily be shared, thus offering the possibility for cooperation skills improvement (Mayes & De Freitas, 2004). This is very important since formal learning environments, such as faculties or campuses originally, amongst other reasons, exist to promote chance encounters and awareness of other people's activities (Morken & Divitini, 2005).

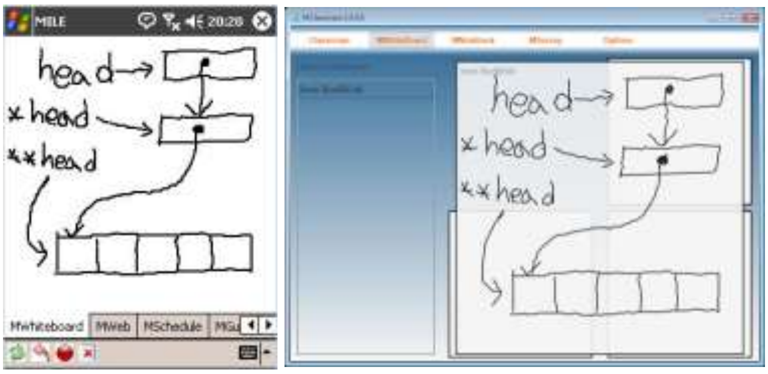


Figure 13. Teacher's view (right hand side) of a student's drawing on MWhiteBoard module (left hand side)

CONCLUSIONS AND FUTURE PLANS

The chapter presented a model for context - aware mobile learning and a system named MILE as an example of its implementation. Context – aware mobile learning utilizes the characteristics of both mobile learning systems and context – aware systems in order to enable learning with mobile devices in contexts.

In addition to being used by individuals outside of their living and working environments, mobile technology is used by certain professions in order to acquire contextualized knowledge while on the move. Mobile learning tries to make learning independent of the location it happens at. In that way learning becomes more personal, less formal and better adapts to learners. Mobile learning is ambient as well. That means it better adapts to students and the environment. The art of developing applications that better adapt to their users is called context – awareness and should be used as a first class concept when designing application for mobile clients.

Following Wang's definition of CAML (context aware mobile learning), MILE possesses basic contextual dimensions called identity, spatio-temporal and the facility dimension. In order to support collaborative activities, the activity and community dimensions of context – aware mobile learning are utilized. The chapter presented a generalized model for context – aware mobile learning designed to be universally applicable when designing system for teaching and learning with mobile connected devices. In addition to sensors, the lowest level of the generalized model of context-aware mobile learning architecture is composed of widgets encapsulating sensor functionality and enabling outer entities to create subscriptions to sensor events. First order aggregators communicate directly with the widgets and are used as a support to the higher-order aggregators.

Main libraries of the MILE system are the Base Framework, the Device Framework and the Server Framework. The Device Framework and the Server Framework are built upon the Base Framework to provide services to specific parts of the system. The Desktop Framework is used by the applications for desktop computers; the Device Framework for the client application and its applicative modules while the Server Framework provides the base for the Contextual Information Service, Event Service, System and Applicative Services. Since MILE utilizes location-aware data special attention was given to the users' need for privacy. Using the client – side privacy widget, users can switch to the private mode and still use some system resources.

Significant effort was invested into the mechanisms for utilizing network resources in order to achieve adequate communication mechanism in the system. The Distributed Events Protocol was created to disseminate events within a learning environment based on mobile devices. Different synchronization mechanisms had to be implemented since the UDP protocol had to be utilized for communication. Orientation on wireless networks presented a problem since mobile devices tend to change their IP addresses frequently.

Applicative modules for mobile devices were implemented as self-contained and pluggable to enable their easy installation and de-installation. They communicate with the rest of the system through an object oriented programming interface in order to maximize reusability and establish firm module boundaries. Although the code was designed to be reusable, special implementation of the client side applicative modules had to be done for each mobile device platform used.

In order to demonstrate the possibilities of contextual mobile learning, few client applicative modules were implemented. MNoteBook module is designed in order to support the didactical approach to mobile learning, while MCollaboration and MWhiteBoard modules provide support for mobile computer supported collaborative learning.

REFERENCES

Ally, M. (2004). Foundations of Educational Theory for Online Learning. In: T. Anderson & F. Elloumi (Eds.), Theory and Practice of Online learning. Athabasca University.

Alonso, F., López, G., Manrique, D., & Viñes, J. M. (2005). An instructional model for web-based e-learning education with a blended learning process approach. British Journal of Educational Technology, 36 (2), 217-235.

Bersin, J. (2004). The Blended Learning Handbook. San Francisco: Wiley.

Boticki I., Mornar, V., & Hoic - Bozic N. (2006). Introducing Location - Awareness into a Learning Environment Supported by Mobile Devices. Pages 267-274.

Boticki, I., Mornar, V., & Andric, K. (2006). Supporting and enhancing teaching and learning with mobile devices. CD-ROM edition. ISBN: 3-89958-195-4.

Boud, D., & Feletti, G. (2001). The Challenge of Problem Based Learning. London, Kogan Page.

Brown P. J. (1996). The e-stick document: A framework for creating context-aware applications. Presented at the Proceedings of the Electronic Publishing. Pages 259-272.

De Freitas S., & Levene M. (2003). Evaluating the development of wearable devices, personal data assistants and the use of other mobile devices in further and higher education institutions (JISC technical report). United Kindgom, JISC.

Dey A. and D. Abowd D. (2000). Towards a better understanding of context and context – awareness. Pages 304-307.

Dey A., Abowd D., & Salber D. (2001). Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction (HCI) Journal, 16(2-4), 97-166.

European Commission (1991). Communication from the commission to the council and the European parliament: the e-Learning action plan. http://eur-lex.europa.eu/LexUriServ/site/en/com/2001/com2001_0172en01.pdf.

Griswold, W. G., Shanahan, P., Brown, S. W., Boyer, R., Ratto, M., Shapiro, R. B., & Truong, T. M. (2004). Activecampus: experiments in community-oriented ubiquitous computing. Computer , 37 (10), 73-81.

Hazas M., Scott J., & Krumm J. (2004). Location-Aware Computing Comes of Age. IEEE Computer Magazine, 37 (2), 95-97.

Hoic-Bozic N., Boticki I., & Mornar V. (2005). Interaction and Collaborative support in an Adaptive Web-based LMS AhyCo. Pages 123-230.

Holzinger A., & Errath M. (2007). Mobile computer Web-application design in medicine: some research based guidelines. Universal Access in the Information Society, 6, 31 – 41.

Hsi S., & H. Fait H (2005). RFID enhances visitors' museum experience at the Exploratorium, Communications of the ACM, 48(9), 60-65

IEEE Std 802®-2001, Media Access Control (MAC). http://standards.ieee.org/getieee802/download/802-2001.pdf.

Jackson B. (2006). History of VoIP. University of Texas at Dallas.

Kaleidoscope Network of Excellence (2006). Mobile Learning Initiative Big Issues in Mobile Learning (Report). University of Nottingham. http://mlearning.noe-kaleidoscope.org/public/news/KALEIDOSCOPE%20REPORT_07_Big_Issues_In_Mobile_Learning.pdf.

Kukulska-Hulme A. (2007). Mobile Usability in Educational Contexts: What have we learnt?. The International Review of Research in Open and Distance Learning, 8(2). 1-16.

Kukulska-Hulme A., & Traxler J. (2005). Mobile learning: A handbook for educators and trainers. Routledge, United Kingdom.

Markett, C., Sanchez, A. I., Weber, S., & Tangney, B. (2006). Using short message service to encourage interactivity in the classroom. Computers & Education , 46 (3), 280-293.

Massey A. P., Ramesh V., & Khatri V. (2006). Design, development, and assessment of mobile applications: the case for problem-based learning. IEEE Transactions on Education, 49(2), 183-192.

Mayes T., & De Freitas S. (2004). Review of e-learning theories, frameworks and models (JISC e-Learning Models Desk Study). United Kingdom. http://www.elearning.ac.uk/resources/modelsdeskreview/.

McCormack C., & Jones D. (1997). Building a Web-Based Education System, New York, Wiley.

Microsoft patterns & practices Developer Center. Microsoft Developer Network. http://msdn.microsoft.com/en-us/practices/default.aspx.

Morken E. M., & Divitini M. (2005). Blending mobile and ambient technologies to support mobility in practice based education: the case of teacher education. Online edition.

Naismith L. (2005). Literature Review in Mobile Technologies and Learning (Report 11, Futurelab Series). Bristol, United Kingdom, Futurelab. http://www.futurelab.org.uk/resources/documents/lit_reviews/Mobile_Review.pdf.

Pascoe, J., Ryan, N., & Morse, D. (2000). Using while moving: Hci issues in fieldwork environments. ACM Trans. Comput.-Hum. Interact. , 7 (3), 417-437.

Peters O. (1999). A pedagogical model for virtual learning spaces. Articles on flexible learning and distance education. Fernuniversitat Hagen. http://www.tbc.dk/pdf/peters-a_pedagogical_model.pdf.

Proctor N., & Burton J. (2003). Tate Modern multimedia tour pilots 2002-2003. Pages 54-55.

RFC 675. Specification of Internet Transmission Control Program, December 1974 Version. http://tools.ietf.org/html/rfc675.

RFC 4120. The Kerberos Network Authentication Service (V5). http://www.ietf.org/rfc/rfc4120.txt.

Sakkopoulos E., Lytras M., & Tsakalidis A. (2006). Adaptive mobile web services facilitate communication and learning Internet technologies, IEEE Transactions on Education, 49(2), 208-215.

Sharples M., Taylor J., & Vavoula G. (2005). Towards a theory of mobile learning. Online edition.

Thorne, K. (2003). Blended Learning. London, Kogan Page.

Yuan-Kai W. (2004). Context Awareness and Adaptation in Mobile Learning. Pages 154-158.