Journal of Communication and Computer 13 (2016) 64-76
doi:10.17265/1548-7709/2016.02.002

Virtual Education Space

Vladimir Valkanov, Stanimir Stoyanov and Veselina Valkanova
Faculty of Mathematics and Informatics, University of Plovdiv, Plovdiv 4003, Bulgaria

Abstract: The wide usage of the Internet and its gradual evolution into an Internet of Things is a foundation for the rapid development of cyber-physical social systems. Those kinds of systems could introduce new approaches and scenarios for solving complex problems in the field of electronic education. The paper presents a VES (virtual education space) aiming at delivery of electronic education services and teaching content, personalized and customized for each individual user. The space supports the education process at the University of Plovdiv, Bulgaria. Problems related to the implementation and modeling of the space are considered as well.

Key words: E-learning, virtual education space, intelligent agents, BDI architecture, personal assistants.

1. Introduction

In recent years the interest towards electronic education has been growing stronger. As a result of that many universities have developed and implemented their own systems for electronic and long-distance education. In line with this trend a DeLC (distributed e-Learning centre) project was implemented in the Faculty of Mathematics and Informatics aiming at the development of an infrastructure for context-aware delivery of electronic education services and teaching content, personalized and customized for each individual user [1, 2].

DeLC is a reference architecture, supporting a reactive, proactive and personalized provision of education services and electronic content. The DeLC architecture is modeled as a network which consists of separate nodes, called e-Learning nodes. Nodes model real units (laboratories, departments, faculties, colleges, and universities), which offer a complete or partial educational cycle. Each e-Learning Node is an autonomous shost of a set of electronic services. The configuration of the networked ges is such as to enable the access, incorporation, use and integration of electronic services located on the different e-Learning Nodes. The e-Learning Nodes can be isolated or integrated in more complex virtual structures, called clusters. Remote e-Service activation and integration is possible only within a cluster. In the network model we can easily create new clusters, reorganize or remove existing clusters (the reorganization is done on a virtual level, it does not affect the real organization). For example, the reorganization of an existing cluster can be made not by removing a node but by denying the access to the services offered by it. The reorganization does not disturb the function of other nodes (as nodes are autonomous self-sufficient educational units providing one or more integral educational services).

Wired and wireless access to the e-Learning services and teaching content have been implemented in DeLC. The current DeLC infrastructure consists of two separate education clusters [3]. The first one, known as MyDeLC, delivers educational services and teaching content through an educational portal. The second one provides mobile access to services and content over an extended local network called Info Stations.

VES (virtual education space) is developed as a successor to DeLC accounting for two important tendencies in the development of Internet and Web. The broad usage of the Internet and its steady transformation into a network of objects [4], as well as

Corresponding author: Stanimir Stoyanov, Prof., research fields: artificial intelligence, software engineering

the globalization of cyberspace, are a foundation for the rapid development of cyber-physical social systems which will lead to essential technological, economical and sociological consequences in the following years. The phrase cyber-physical system is used to specify a tight integration and coordination between computational and material resources where a compact integration between calculation, communication and control exists as well as an interaction with the environment in which they are situated [5]. For a number of application areas it is necessary to measure the presence of the human and social dimensions within those spaces. We have reached the point where the social and human dynamics become an indelible part of the cyber-physical space so that the inclusion of the term "social" is completely reasonable. One logical consequence is the notion of pervasive intelligent spaces, where humans and objects interact intelligently among themselves in a way known as anywhere-anytime-anyhow. The spaces become intelligent when they are capable of monitoring what is happening inside them, have the ability to model their own behavior and to operate on the basis of their own decisions as well as interact with the inhabiting communities. Apparently, besides residents, these spaces require the buildup of a suitable information infrastructure.

Those types of spaces can introduce new approaches and scenarios for solving complex problems in the field of electronic education. An important tendency in electronic education is that it is based on the integrated character of the highly technological world that people live and study in. A basic priority is the development of education spaces via integrating different technologies which engage the students and increase the interest towards the learning process in ways that were heretofore impossible; create new opportunities for education and teaching, improve and broaden the interaction with local and global communities [6]. Education spaces, both physical and virtual, are planning environments where different forms of integrated education are performed. They connect the school, home and educational community by increasing and supporting a flexible education outside the boundaries of the school buildings and outside the usual school days. Those spaces can also direct the making of strategic decisions in schools, government and educational institutions.

A second important tendency that appears with the idea of transforming DeLC into a virtual education space is the creation of semantic web. The semantic web is a development of the current syntactic web where the infrastructure delivers a model of machine-comprehensible data. These data are stored separately but can easily be integrated if necessary. The idea for semantic web was first introduced by Tim Berners-Lee et al. in Refs. [7, 8]. Ideas for the usage of semantic web in the electronic education are presented in Ref. [9]. At the current moment using the capabilities of semantic web in electronic education is the subject of strong scientific interest.

The first ideas for the VES are presented in Ref. [10]. A detailed characteristic of the space is given in Ref. [11]. In this publication, we make a general characteristic and present the architecture of the VES. We also discuss problems related to the implementation and modeling of the space.

2. VES Overview

2.1 Basic Characteristics of the VES

VES is an intelligent space. In accordance with Refs. [12, 13] an intelligent space is an environment that can continuously monitor what is happening in it, can communicate with its inhabitants and neighborhoods, can make related conclusions and decisions and act on these decisions. In comparison to DeLC, an intelligent education space will support more effectively the process of blended learning, integrating electronic forms of education with the real learning process.

VES is context-aware. According to Ref. [14] context is any information which can be used to characterize the situation of an identity. By “identity” we can designate a man, a place or an object which are viewed as meaningful for the interaction between user and application, which includes themselves. In accordance to the definition of context, Dei defines that a system is context-dependent if it uses contexts to deliver significant information and/or services, and the importance depends on the user’s tasks [14, 15]. In our case context-dependency is the ability of a system to find, identify and interpret the changes (events) in its environment and depending on their nature to undertake compensating actions. The main compensating actions (attributes for context-dependency) are personalization and adaptation. Personalization is the system’s ability to adapt to individual features, desires, intentions and goals of the users. Adaptation is the system’s ability to adapt to the remaining context features such as area of knowledge, school subject, types of devices used by the end-users.

VES is scenario-oriented. From the user’s point of view, the space is a set of separate e-learning services and educational scenarios provided for the use through education portal DeL Corpersonal assistants. Scenarios are implemented by corresponding work flows rendering an account of the environment’s state. Thus it is possible to take into account various temporal characteristics (duration, repetition, frequency, start, and end) of the educational process or events (planned or accidental) which can impede or alter the running of the current educational scenario. To deal with emergencies (earthquake, flood, fire) there are defined emergency scenarios which are executed with the highest priority.

VES is a controlled infrastructure. Access to the space’s information resources is only possible through the so-called “entry points.” The personal assistants operate as typical entry points while the education portal of DeLC is a specialized entry point; a user has to be in possession of a personal assistant or to use the portal to be able to work in the space.

2.2 VES Architecture

The VES architecture contains different types of components. Assistants play an important role in the space. Three types of assistants are supported in the space (Fig. 1). The PAs (personal assistants) have to perform two main functions providing the needed “entry points” of the space. Firstly, they operate as an interface between their owners and the space and if necessary, carry out activities related to personalization and adaptation. Secondly, they interact with other assistants in the space in order to start and control the execution of the generated plans. In certain cases they operate as an intermediary for activation of scenarios or services. The personal assistants will be usually deployed over the users’ mobile devices. The specialized assistants are usually located on the server nodes of the VES, known as OpAs (operatives).They support the execution of the plans generated by the personal assistants; therefore they implement suitable interfaces to the available electronic services and data repositories. Operatives serve two subspaces, known as D-Subspace and A-Subspace respectively. Guards are special assistants which are responsible for safety and the efficient execution of the plans in the space. These are usually intelligent devices that react to various physical quantities in the environment, e.g. smoke, temperature, humidity. The guards act as an interface between the physical and the virtual world in the space.

2.3 Assistants as Intelligent Agents

VES is “populated” only by active components known as assistants. Each assistant has to play a role in accordance with its delegated responsibilities. The responsibilities (tasks) can be implemented or delivered by external electronic services. The electronic services themselves cannot be separate operational components of the space because they are

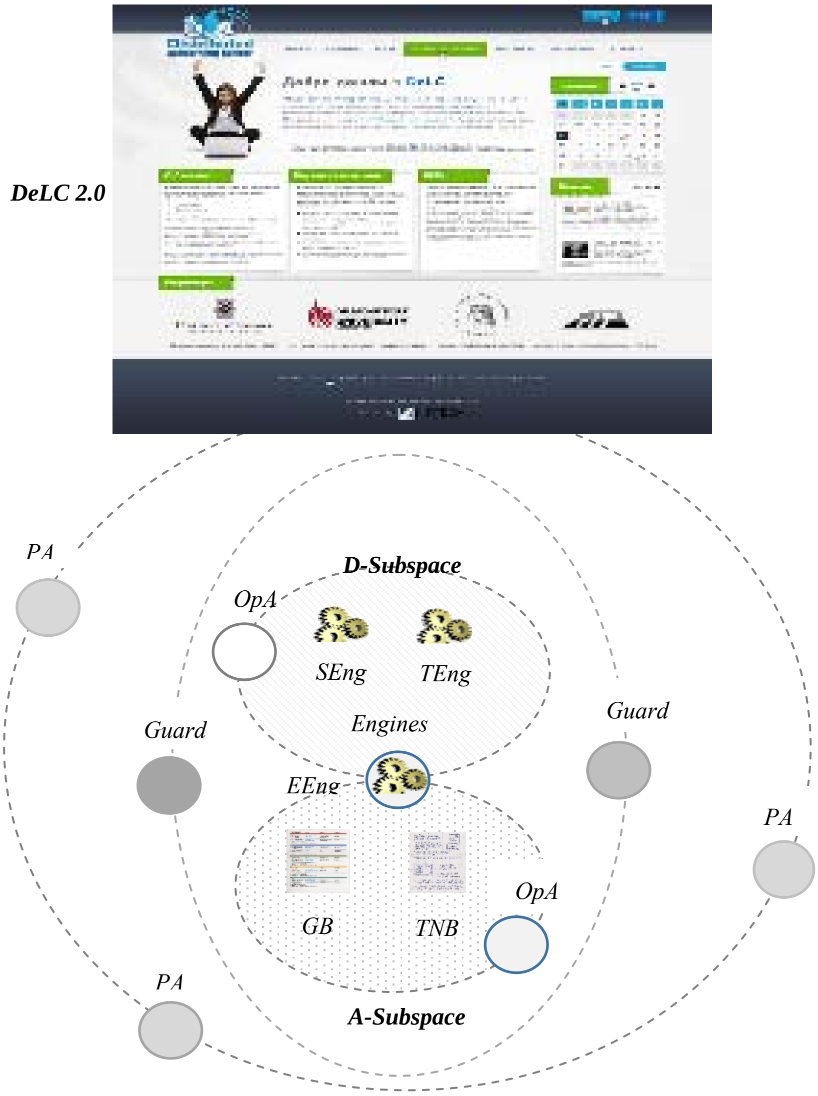

Fig. 1 VES architecture.

suitable for implementation of business functionality but are static without having the properties to be context-aware and intelligent.

The assistants, implemented as intelligent software agents, constitute the kernel of the space. The assistants are active context-aware intelligent components that support the planning, organization and implementation of the educational process. The basic features of the intelligent assistants are:

- *Autonomy* – they operate without the direct intervention of humans or other agents, they have control over own actions and internal states;
- *Reactivity* – they perceive their environment maintaining continuous contact with this environment and respond to changes that occur in it;
- *Proactivity* – they not only act in response to their environment. They are able to be proactive exhibiting a goal-driven behavior;
- *Sociality* – they are able to interact and cooperate with other assistants via some communication language (e.g. ACL [16]).

Furthermore, the space assistants are implemented as limited rational agents which mean the following:

- They have only partly control over the space;
- Operate with limited capacity (resources) on the planning, forecasting, selection and execution of actions;

- In accordance with the task completed they are able to assess their behavior and based on these evaluations to measure the effectiveness of their actions. Significant in the "bounded rationality" is finding meaningful solutions for efficient use of available resources. The scope of the action of a rational agent can be defined as optimal success according to its current knowledge and skills, limited resources and limited time. A model known as "practical considerations" can be applied for decision-making by rational agents. A rational agent holds a practical consideration performing two steps:
- Deliberation – at this stage, the agent decides what to do (what purpose it wants to achieve) using its mental states;
- Planning (means-ends reasoning) – at this stage, the agent decides how to meet the goal.

The operation of a rational agent is presented with the help of its life cycle where deliberation and planning is fitted with sensors and effectors. BDI (Belief-Desire-Intention) architecture [17] presented a human-like model for bounded rationality which is usually used for the development of rational agents. In the deliberation step three mental states of the agent play an important role:

- Beliefs represent perceptions of the agents about their environment;
- Desires represent the agents' desires or tasks that have not been converted into intentions;
- Intentions are committed desires.

In the means-ends step an agent outlines the process for addressing how to achieve a goal using existing resources (also known as planning). The planning module of the agent takes as input a structure consisting of:

- Goal, i.e. the current intention of the agent;
- Current state of the environment, i.e. the beliefs of the agent;
- Possible actions of the agent.

The result is an action plan for achieving the goal.

3. Personal Assistants

The main purpose of PAs (personal assistants) is to assist the personalization of the space. They perform mainly the following functions:

- They serve as an entry point in the space – typically by using the space where the users identify themselves through their own PA;
- They assist the different users in their usage of the space's information resources – they have different functionalities depending on the user type.

3.1 PA's Life Cycle

The personal assistants operating in the space are intelligent agents with bounded rationality. PA's life cycle is implemented in accordance with such proposed in Ref. [18] (Fig. 2). During the deliberation a personal assistant tries to determine its current goal (intention) with the help of the three mental states (Beliefs, Desires, and Intentions) defined as follows:

- Beliefs – in different scenarios with the participation of PA beliefs can be different things. In the PA's realization the basic beliefs are data, time, location and student profile;

```
/* Initialization */
$Profile \leftarrow B_0; PC \leftarrow D_0(Profile);$
$Beliefs \leftarrow Profile; Desires \leftarrow PC;$
$Intention \leftarrow I_0;$
while true do
    /* Deliberation */
    $p \leftarrow see(Environment);$
    $Beliefs \leftarrow brf(Beliefs, p);$
    $Desires \leftarrow update(Desires, B, I);$
    $I \leftarrow select(Desires, B, I);$
    /* Planning */
    $\pi \leftarrow plan(B, I, Ac);$
    $run(\pi);$
end-while
```

Fig. 2 PA's life cycle.

- Desires – in the suggested life cycle desires have a key role. Desires are presented as a personal calendar of a user in the space;
- Intentions – in this case they are usually select events from the personal calendar on which the PA focuses at a certain time;
- Plans – the plans are related with the purpose and permissions of PA.

In our model the mental states are presented as events. An event is defined at the level of the applicable area and is loaded with the semantics of terms from that area; thus, events are holding a lecture, participating in a seminar, holding an examination, exercises. Those events can be characterized through different attributes.

3.2 Events

The unified presentation of the mental states and relations between them will ease the development of a general model of an agent's deliberation. In the suggested architecture for a personal assistant the events perform this function.

An event is a basic concept within the VES. However, a universal definition has not been reached, as multiple theories exist concerning events; e.g. in computing, an event is an action or occurrence detected by the program that may be handled by the program [19]; in philosophy, events are objects in time or instantiations properties in objects [20]; in probability theory, an event is a set of outcomes of an experiment to which a probability is assigned [21]; an event in the UML (unified modeling language) is a notable occurrence at a particular point in time [22].

In the space, events take place as in the real world as well as in the virtual one bridging the both. However, all these events have to be represented in the virtual world (assistants) to make them amenable to usage from the assistants. An event can be presented as an atomic identity which can only be identified but not attributed, or the event can be presented as an identity which can be attributed by allowing the identification and working with the different attributes. In the space we accepted the model given in Ref. [23]. The presented model is a six-faceted model that aims at establishing acommon foundation for a wide diversity of applications and addresses several elementary aspects of event description. These aspects are briefly characterized as follows [23]:

- Structural – events are a modeling concept applicable at many different levels of abstraction. Events can be combined in many different ways to define other events, which can also be related to other events, thus creating complex structures and hierarchies of events [24];
- Informational – an event model should provide information about the events that occur (e.g. event type) and should support a taxonomical organization of event types. The way for creating complex events and hierarchies is determined by the area of application;
- Temporal – the temporal characteristics of the events have to be taken into account because the events are inherently related to the concept of time;
- Spatial – a common event model should also show location awareness and support different ways of capturing the spatial aspect in an event's description;
- Causal – in many applications the users are interested in the chain of events;
- Experiential – various applications offer users engaging ways of exploring and experiencing a course of events to let them gain insights into how the events evolved.

3.3 PA's Initialization Phase

Even though it is given within the PA's life cycle, in fact the PA is created during this phase as an instance of a reference GPA (generic personal assistant). This is accomplished according to the following scheme:

- GPA interacts with the educational portal DeLC

2.0 – DeLC 2.0 supports the actual school curriculum (*D_0*) for conducting the learning process. Furthermore, within the portal the students'(*B_0*) personal information is kept such as faculty number, academic major, academic year. During this interaction initially the PA user's profile is generated (*Profile B_0*). Afterwards this profile is used as a selector for generating the personal calendar (*PC ← D_0 (Profile)*).

- GPA interacts with the user – the PA users, interacting with GPA, can redact their own profile (*Beliefs Profile*) and their personal calendar (*Desires ←PC*).
- GPA activates the newly generated PA – PA initializes its current intention (*Intention I_0*) and is now responsible for the actual maintenance of its mental states (deliberation phase).

3.4 PA's Deliberation Phase

In the deliberation phase a central role is played by desires, presented as a personal calendar of the user. Thus for instance, the personal calendar of a student would include all activities that she/he has to perform in order to successfully complete the school year. Those can include attending lectures, participating in seminars, taking examinations. All those activities could be interpreted as past, current, or potential events according to the situation.

The actual goal(s) is determined via an algorithm (*select(Desires, B, I)*) for selecting the personal calendar(*Desires*), which has to answer the following question: "Which event (or events) from the personal calendar require the user's response with a suitable action in the current moment?".

The role of Beliefs is to determine definitively the actual access to the personal calendar. The goal of the actual access is to identify a desire (or a number of desires) from the calendar, a candidate for becoming the agent's actual goal. Afterwards, this desire is transformed into an actual goal (*Intention*). In accordance to the actual Beliefs there are two types of desires: ones that can be transformed into Intention and ones that only remain Desires (e.g., an absence from a lecture, seminar or an examination). In our model there are two types of basic Beliefs; they are the current date and current time. Independently from the type of the event which was perceived by the agent, they play an important role in the implementation of the actual access to personal calendar.

For the selection's accurate operation the following preparatory actions are required from the PA:

- Actualization of Beliefs (*Beliefs ← brf(Beliefs, p)*) in accordance with the actual perceptions (*p*);
- Actualization of the personal calendar (*Desires ← update(Desires, B, I)*) – in this case the actualization of the personal calendar is usually initialized as a reaction to messages received by certain Operatives, Gards, or DeLC 2.0.

3.5 PA's Planning Phase

After the actual intention has been determined during the deliberation phase, the PA has to make a plan for its achievement. In our model the planning is performed in dependence with the type of the current Intention. In the actual realization these are: informing about changes in the personal calendar (e.g., changing a date, time or place of a lecture) or generating and sending different warnings or reminders for upcoming events or missed assignments by the users in accordance with the assignments in their personal calendars.

4. Operatives

Operatives are the second large group of active components in the space. They are usually situated on the space's servers. Operatives (realized as intelligent agents) are not suitable software components for delivering business-functionality. The functionality is realized as electronic services which in their nature are static and cannot operate as separate components in the space. For that reason a suitable interface operative is supplied for each service, which makes the respective service an active component by allowing it to interact with other components in the space. In accordance with the functions that the operatives perform, the space can be viewed as containing two subspaces called DiLib-Subspace and Admin-Subspace.

4.1 D-Subspace

The main goal of the operatives functioning in the DiLib-Subspace, is to secure the execution of the educational scenarios which are directly related to the education process. When an educational scenario has to be realized, the necessary operatives secure the suitable work stream. The operatives of that subspace realize interfaces to three components supplied to the space by DeLC 2.0 –*SEng* (SCORM 2004 engine), *TEng* (test engine), and *EEng* (event engine) (shared by both subspaces).

The operatives' environment in that subspace is mainly comprised of digital libraries. *The digital libraries* are a specialized storage where mainly educational content is stored. Using the SCORM 2004 standards (for presenting educational content) and QTI 2.1 requires a certain structure of the digital libraries. In them they also store specifications for the educational scenarios.

The digital libraries are created and actualized mainly with the help of specialized editors delivered to the users by the digital interface. When necessary, their content can be dynamically actualized by the serving operatives.

4.2 A-Subspace

Admin-Subspace secures all activities related to the organization, control and documentation of the education process. In the administrated database is stored all the necessary useful information for planning, organizing, protocoling and documenting the educational process such as school plans, programs and schedules, protocols from examinations, grade books.

One example for organization of the school process is the preparation of the students' personal assistants. What is significant here is the creation of individual student calendars (desires of the personal assistants serving them). The individual calendars are generated automatically by the school curriculum for a school term or a school year. From the school curriculum certain events can be derived which are related to the student's immediate learning activity such as lectures, seminars, examinations. Afterwards, the calendar can be completed with events of extracurricular character such as a birthday party, meeting friends, hobbies.

Two Operatives play an important role in that subspace – the electronic grade book and the teacher's notebook:

- *GB* – the student's grade book stores and analyzes information on the success rate of students in all the studied courses. It is currently being developed in accordance with the Grade Book specification of the Common Cartridge standard [25].
- *TNB (teacher's notebook)* – it is designed for the analysis of the success rate of students in a particular course of studies. In addition, it helps the teacher to organize his/her duties during the current education period.

4.3 Interaction between the Two Subspaces

The two subspaces interact intensively for securing blended learning. An example for such an interaction is the scenario for protocoling the students' electronic testing (Fig. 3). The PAS (student's personal assistant), interacting with the PAT (teacher's personal assistant), determines for executing an educational scenario for electronic testing. CO (catalog operative), operating within the D-Subspace, analyzes the scenario and determines the necessary work flow, i.e., the IO (interface operative) and CAO (content access operative) agents serving the scenario's execution. Since the activation of the Test Engine is realized through the client layer of the space, the PAS is used for that goal. The results from the test are written in the student's book of the tested student which causes a

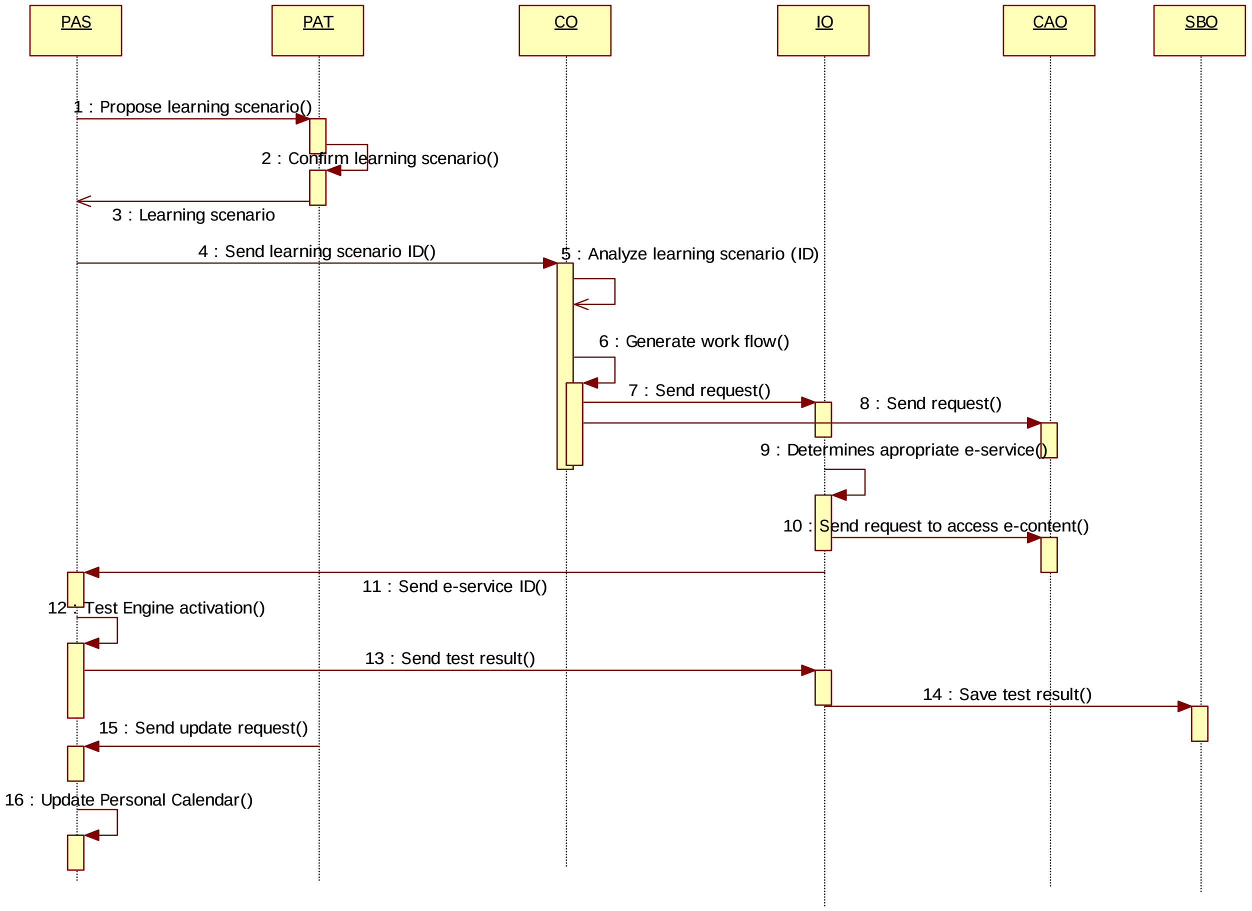
PAS
PAT
CO
IO
CAO
SBO
1 : Propose learning scenario()
2 : Confirm learning scenario()
3 : Learning scenario
4 : Send learning scenario ID()
5 : Analyze learning scenario (ID)
6 : Generate work flow()
7 : Send request()
8 : Send request()
9 : Determines apropriate e-service()
10 : Send request to access e-content()
11 : Send e-service ID()
12 : Test Engine activation()
13 : Send test result()
14 : Save test result()
15 : Send update request()
16 : Update Personal Calendar()

Fig. 3 D–A-Subspace interaction.

change in the environment and the activation of the SBO (student book operative). Depending on the results (e.g., a failed test), the SBO informs the PAT which in turn requires from the PAS an actualization of the student's PC (requesting a date for re-taking the examination and on confirmation, storing it in the PC).

4.4 Operatives' Environment

Context-Awareness and intelligent behavior cannot be achieved only by the presence of intelligent agents. These have to be also supported in the agents' environment, i.e. intelligently structured data are also required. Under "intelligent" we understand suitably structured data which can be stored separately and when necessary (executing complex queries), easily integrated [26].

The environment of the assistants operating in the D-Subspace, is built in relation to the requirements of the standards SCORM 2004 and QTI 2.1 for structuring the necessary repositories. The administrative information in the A-Subspace is stored in relational databases. A flaw of these repositories and data bases is the absence of semantics. For that reason in the assistants' environment different educational ontologies are developed. Using semantic technologies we intend to achieve the following benefits:

- Education focused on the student – the teaching material, possibly from different authors, can be related to commonly agreed ontologies. Personalized courses can be developed through semantic queries. The teaching material can be sought and derived from the context of actual problems in accordance with the decisions of the learner;
- Flexible access – the knowledge can be

accessible in any order that the learner desires. The circumstances will be determined by suitable semantic annotations. Support for non-linear access;

- Integration – An integrated platform for the business processes of the organizations. The educational activities can be integrated in those processes.

The ontologies are also an effective mechanism for realizing a shared comprehension. In electronic education it is suitable to build the following storages for knowledge (ontologies) [27]:

- Ontology of the teaching content – containing basic concepts of the area in which the education is performed, and the relations between them;
- Pedagogical ontologies – for solving pedagogical problems related to the comprehension of the teaching material. For instance, the teaching material can be classified as: lessons, exercises, illustration examples, solutions for problems with a high degree of difficulty;
- Structural ontologies – define the logical structure of the teaching material and typical knowledge of that type includes hierarchical and navigational relations.

5. VES Implementation

DeLC 2.0 operates as a special entry point in the space allowing access to the space's resources out of the personal assistants. DeLC 2.0 is built as a dynamic web application distributed in two main areas – an education portal operating as a specialized user interface and server side (Fig. 4). Both areas communicate by using pure HTTP requests, RESTful services and Web Sockets. An application in the browser consists of a pure HTML 5 and CSS 3 implementation combined with dynamics provided by JQuery and responsiveness delivered by Bootstrap.

At the server side, the core services provided by DeLC 2.0 include the minimal functionality needed for the majority of the current web applications: security (implements the security mechanism with per roles, per groups and per users' rules), Web Content (it allows a definition of static web content), and Control Panel (the management of the above services, the plugins, and so on). There are also two additional services that are provided by the platform at a lower level: Message Queue (is used by all the plugins and core services as a message bus for asynchronous communication between the different components) and ORM Layer (an object-relational transformation layer that is used by all components for communication with the database). Furthermore, DeLC 2.0 supports interfaces to D- and A-Subspaces by means of plug-in technology. Each plugin provides unique functionality which is integrated in the portal seamlessly. Currently, SCORM engine, Test Engine,

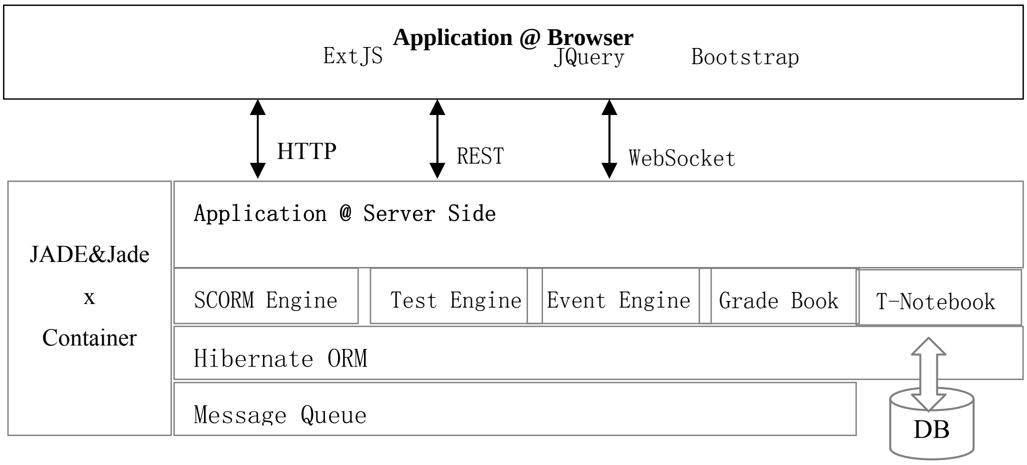

Fig. 4 DeLC 2.0 architecture.

Event Engine, Grade Book, and T-Notebook plugins in a different state of completion are supported by the platform. The JADE [28] & Jadex [29] Container makes possible the creation of agents that can operate in the portal and in this way to ensure interaction with the operative assistants of the space.

For that reason we pay attention to the distributed architecture, which is further determined by JADE-LEAP (Lightweight Extensible Agent Platform) selected technology for implementation of the personal assistants. This is an extension to the basic JADE specifically designed for mobile devices with limited resources [30]. The choice of this technology is motivated by many reasons, some of which are the following:

- The platform is developed entirely in Java, which coincides with the language used for the development of VES and the Android operating system;
- Facilitates the development of multi-agent systems via an intermediate layer that is compatible with the specifications of FIPA [31];
- Provides tools to facilitated bugging and deployment and there is a transparent communication mechanism between the agents through ACL messages;
- JADE supports the use of many additional libraries, one of which provides the important to us BDI architecture, through which we can implement the mental states of intelligent agents, namely autonomy, social ability, reactivity, proactivity and rationality.

In addition to JADE-LEAP ensuring dealing with the mental states of the agents we are integrating JADEX environment. The JADEX reasoning engine follows the BDI model and facilitates easy intelligent-agent construction with sound software engineering foundations. It allows agents' programming in XML and Java and can be deployed on different kinds of middleware such as JADE. The platform allows mixing of agents with different architectures in one application.

JADEX provides also a framework for developing software agents running on the Android platform.

The personal assistants, responsible for interaction with users should provide a convenient and intuitive interface. This is a reason why we have chosen the Android platform. Furthermore, at the present, Android is the most popular mobile platform in the world and supports a great number of different mobile devices. In addition, the Android provides a very powerful development framework for building applications for mobile devices [32]. It automatically adapts the user interface to look its best on every device, while giving the developer as much control as he wants over the whole user interface on different device types. There are convenient tools for developers which offer a full Java IDE with advanced features for developing, debugging and packaging Android applications. The environment offers an emulator of virtual devices that emulates any hardware configuration. The platform uses Java and is open source, which allows working with multiple, external, written by third-party libraries. The Android philosophy is to support an extremely high level of user customization**.**

6. VES Modeling

During the third stage we predict the development of models of the space and experimenting with them. With the help of those models we aim at:

- Checking and studying in-depth the expected behavior of the assistants and the interactions among them;
- Studying and understanding the temporal dependencies in the space;
- Supporting the formalization of structures in the space.

Our goal is to develop a united technological environment for modeling which includes three formal systems (CCA, CS-Flow, Tempura), which will support both the modeling of the entire space and its separate components. Context-awareness requires applications to be able to adapt themselves to the environment in which they are being used such as user, location, nearby people and devices, and user's social situations. The CCA (calculus of context-aware ambient) [33] has been proposed to model context-aware systems. The CCA is a process calculus based on the notion of ambients. CCA is used to model ambient in terms of process, location and capability. In Ref. [34] are demonstrated the capabilities of the calculus for modeling the mLecture, mTest and mTutorial services presented by DeLC.

The context-aware process management in the Virtual Education Space is a significant problem to be solved. We believe a scenario-oriented management is appropriate where various education scenarios can be activated depending on the specific situation in the space. At the same time we are looking for opportunities for formal specification of these scenarios. CS-Flow (short contex secure flow) is a formal interpretable language for presenting of context-aware workflow [35]. Soon we are going to examine the language use in VES.

The most important characteristic of the scenarios is that they are time-dependent, therefore we need a proper formalism to describe them and an interpreting mechanism to correspond with our system. A good approach for developing the required interpreting mechanism is to look for a formal notation which allows on one hand to specify the scenarios with their temporal characteristics and on the other hand allows the existence of a possibility for a program interpretation of those specifications. Usually formal logic-based models are suitable in such situations. Apparently, in this case a formal notation using first order logic is inapplicable, according to time-critical specifics of the system. Thus, the formal notation ITL (interval temporal logic) was chosen for the following reasons [36]:

- ITL is a time-dependent logical lowing the presentation and control of linear and parallel processes. A basic characteristic of this logic is considering time as a discrete sequence of points in time called intervals. Losing the concept of time as an endless notion we can describe different processes in a way similar to the operation of modern computers and software systems;
- For ITL there is an interpreting mechanism and its program realization called Tempura.

In Ref. [37], it is presented the AjTempura, the new version of the Tempura interpreter which is agent-oriented and written in Java. This interpreter can work in a united technological environment for modeling of the space.

7. Conclusion

In this paper, the Virtual Education Space that is an intelligent, context-aware, scenario-oriented and controlled infrastructure is presented. A prototype of the system is used in the real university education. Future tasks are mainly related to the implementation of the full functionality of the space as well as to adaptation of the space for lifelong learning.

Acknowledgment

The authors wish to acknowledge the partly support of the NPD – Plovdiv University under Grants No. NI-FMI-004 "Innovative basic and applied research in computer science, mathematics and pedagogy training", 2015-2016 and No. IT15-FMIIT-004 "Research in the domain of innovative ICT oriented towards business and education", 2015-16.

References

[1] Stoyanov, S. 2005. "From CBT to E-Learning." *Journal Information Technologies and Control* 3 (4): 2-10.

[2] Stoyanov, S. 2008. "An Approach for the Development of Info Station-Basede Learning Architectures." Comptes Rendusde l'Academie Bulgaredes Sciences. 61 (9): 1189-98.

[3] Doychev, E. 2013. "Environment for Provision of E-Learning Services." PhD thesis, University of Plovdiv.

[4] Kevin, A. 2009. "That 'Internet of things', in the Real World Things Matter than Ideas." *RFID Journal* (June).

[5] Wang, F. Y. 2010. "The Emergence of Intelligent Enterprises." *From CPS to CPSS, IEEE Intelligent*

Systems (July/August): 85-8.

[6] Consortium, MCEETYA, Learning Spaces Framework. Australia - New Zealand: MCEETYA Consortium, 2008.

[7] Lee, T. B., Handler, J., and Lassila, O. 2001. "The Semantic Web." *Scientific American* 284 (May): 34-43.

[8] Berners-Lee, T. 2000. *What the Semantic Web Can Represent, W3 org.* Scientific report.

[9] Stojanovic, L., Staab, S., and Rudi, S. 2001. "E-Learning Based on the Sematic Web." WebNet.

[10] Orozova, D., Stoyanov, S., and Popchev, I. 2013. "Virtual Education Space." Presented at the International Conference, Free University of Burgas, 14-15 June, 153-159, ISBN 978-954-9370-99-7.

[11] Valkanova, V. 2014. "Researching the Virtual Learning Space in the Secondary School." Ph.D. thesis, Sofia, ISBN 978-954-322-768-6.

[12] Liu, B. 2007. "Intelligent Spaces: An Overview." Presented at the IEEE International Conference on Vehicular Electronics and Safety, Beijing, 978-1-4244-1266-2.

[13] Wang, F. Y. 2006. "Driving into the Future with ITS." *IEEE Intelligent System* 21: 94-5.

[14] Dey, A. K. 2001. "Understanding and Using Context." *Personal and Ubiquitous Computing Journal* 5: 4-7.

[15] Dey, A. K., and Abowd, G. D. 2000. "Towards a Better Understanding of Context and Context-Awareness." In *Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness*, New York, ACM Press.

[16] Agent Communication Language Specifications, http://www.fipa.org/repository/aclspecs.html.

[17] Rao, A. S., and Georgeff, M. 1995. "BDI Agents: From Theory to Practice." In *Proceedings of the 1st International Conference on Multi-Agent Systems*, 312-19.

[18] Wooldridge, M. 2009. *An Introduction to MultiAgent Systems*. Wiley.

[19] http://en.wikipedia.org/wiki/Event_(computing).

[20] http://en.wikipedia.org/wiki/Event_(philosophy).

[21] Leon-Garcia, A. 2008. *Probability, Statistics and Random Processes for Electrical Engineering*. Upper Saddle River, NJ: Pearson.

[22] http://en.wikipedia.org/wiki/Event_(UML).

[23] Westermann, U., and Jain, R. 2007. "Toward a Common Event Model Formula Timed Applications." *IEEE Multi Media* 14: 19-29.

[24] Rafatirad, S., Gupta, A., and Jain, R. 2009. "Event Composition Operators: ECO." In *EiMM 2009: Proc. 1st ACM Int.* Presented at the Workshop on Events in Multimedia, New York, NY: ACM.

[25] IMS Common Cartridge Specification, https://www.imsglobal.org/cc/index.html.

[26] Allemang, D., and Hendler, J. 2011. *Semantic Web for the Working Ontologist*. Elsevier, ISBN: 978-0-12-385965-5.

[27] Antoniou, G., and Harmelen, F. V. 2004. *A Semantic Web Primer*. The MIT Press.

[28] Bellifemine, F., Caire, G., and Greenwood, D. 2007. *Developing Multi-Agent Systems with JADE*. Wiley.

[29] Jadex Active Components, https://www.activecomponents.org/bin/view/About/New+Home.

[30] JADE-LEAP: http://sharon.cslet.it/project/jade.

[31] Foundation for Intelligent Physical Agents: http://www.fipa.org.

[32] Android for Developers: http://developer.android.com/about/index.html.

[33] Siewe, F., Zedan, H., and Cau, A. 2010. "The Calculus of Context-Aware Ambients." *Journal of Computer and System Sciences*, In Press, Corrected Proof.

[34] Sammarraie, M. H. A. 2011. "Policy-based Approach for Context-aware Systems." Ph.D. thesis, Software Technology Research Laboratory, De Montfort University, Leicester, UK.

[35] Zedan, H. 2012. *CS-Flow. Computational Model—Linguistic Support*. Internal Report, STRL, De Montfort University.

[36] Moszkowski, B. 1986. *Executing Temporal Logic Programs*. Cambridge: Cambridge University Press.

[37] Valkanov, V. 2013. "Context-Aware Management of eService." Ph.D. thesis, Sofia.