

# Ambient-oriented CCA Modeling of Intelligent Wheelchair in Virtual Learning Space

T. Glushkova

**Key Words:** Ambient; Calculus of Context-aware Ambients (CCA); VelSpace.

**Abstract.** This paper presents a Calculus of Context-aware Ambients (CCA) Model of Services for Students with Mobility Problems in the University Virtual Learning Space. These students will be represent in the system through their intelligent wheelchairs that will communicate with the learning environment to provide them the appropriate learning resources and services as well the best-suited route for moving in the real campus area.

---

## 1. Introduction

As we know, in the next years the technologies will transform the entire structure of world economy, communications, as well ways of interacting between people and machines in the real and the virtual world. The experts believe that the Fourth Industrial Revolution will determine the rapid development of wide-ranging fields such as Artificial Intelligence (AI), Robotics, the Internet of Things and so on [13].

The Internet of Things is a new paradigm for distributed next-generation systems, where electronic devices remain in the background in everyday consumer activities [1]. According to this paradigm, computing is based on a number of small devices and sensors that capture and process environmental parameters, thus linking virtual space to the real world.

In the article, we will focus on creating a context-aware model [2], [3] to teach students with mobility problems attending university campus with their intelligent wheelchairs. Sensors of the wheelchair will capture changes in environmental parameters and provide appropriate information, services and educational resources through two-way communication with the virtual educational space. For these students, besides the standard ones, we will describe and model and some specific services related to their movement and positioning in the physical space of the university campus.

## 2. Related Works

Most of the existing eLearning and Distance Learning systems, although addressing a number of issues related to the deployment, searching and delivery of eLearning content, also have certain shortcomings. These disadvantages are related to not enough effective support for students in the course of their training. In order to respond to the ev-

er-increasing requirements of the different learner groups, we need context-sensitive intelligent environments, which require solving a number of problems and challenges, such as: personalization of the learning process [5]; adapting the educational services to different learning courses; adaptation of learning content to students' knowledge, goals and expectations [4], as well as to the type of their access device; closer integration of the virtual learning space into the real physical world in which the learning process is de facto carried out. In this sense, it is necessary for these learning spaces to be able to observe what is happening inside and around them and to be able to adapt their behaviors to these changes. In this direction are the changes that we design and implement in the Virtual Learning Space VelSpace [11] in Faculty of Mathematics and Informatics of Plovdiv University "Paisii Hilendarski". The space have to ensure realization of different scenarios and processes as creating, publishing and sharing e-learning resources; tracking the learning outcomes of each student by intelligent student's book and a lecturer's notebook; the exchange of messages between the various participants in the learning process; the holding of virtual meetings, seminars, consultations, exams, and so on. In this article, we will concentrate only on modeling of some processes and services, related to the education of students with motor problems.

The modeling and prototyping of the different scenarios in the educational space is essential to optimize the services provided before their realization. We will use the notation Calculus of Context-aware Ambients – CCA [14] for modeling scenarios. CCA builds the Calculus of Boxed Ambient (CBA), which inherited the Ambient Calculation (AC) mobility model [12].

## 3. Approach

### Architecture of the Educational Space

The learning space has a distributed architecture, the separate components could connected differently, depending on the current scenario (*figure 1*). The Learning Portal has a major coordinating role in the model. Portal architecture is multilayered, each layer being associated with a certain functionality. The data model is realized by development of two virtual subspaces – DiLib SubSpace and Admin SubSpace.

Digital Libraries (DiLib SubSpace) are used to store learning content. E-learning courses are stored in a separate

repository (LecturesRepository-LR) with meta information for each course (type of professional field or specialty, topic, form of education, type of access device, etc.). Similarly is built and the Test Repository (TR).

The Admin SubSpace uses an external administrative repository with information for students (Student Data Base-SDB) as well as a repository of events related to the curriculum, weekly timetable, exam sessions, current events, and so on (Events Repository – ER).

The different kind of stationary or mobile devices through which the student uses the information have different characteristics and can display differently the received information. According to this context, lectures and tests can be developed in different variants – for mobile devices or PCs.

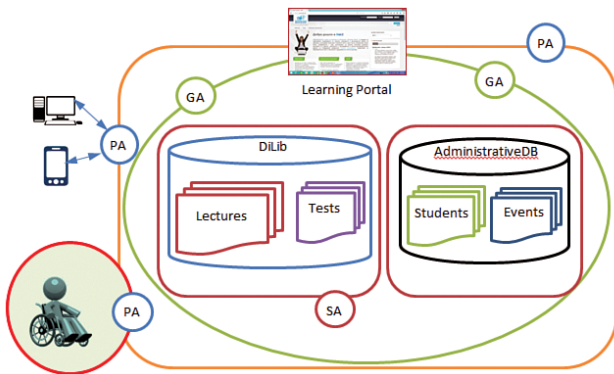


Figure 1. An architecture of the learning space

The educational space is "populated" by assistants who are autonomous components [7], [8]. We expect for them to be proactive, interactive and pro-social, i.e. to communicate with each other through messaging. By their nature, they are intelligent agents who are responsible for the execution of a certain number of services. We will define three types of assistants:

- **PA – personal assistants** that support the work of users in the learning space and deliver "entry points" to it [10].
- **SA – assistant specialists** who provide specialized assistance and services to the users in the execution of their requests in the space. These assistants communicate with each other as provide specific information or service.
- **GA – Guard Assistants**, which have specific security - related functionality during the implementation of educational scenarios.

The student with mobility disabilities participates in the model with his wheelchair, which provides him with a specific support environment. Upon entering in the educational university environment, the PA assistant starts a two-way communication process, providing the student with both standard educational process, providing the student with services connected with his movement in the campus's physical environment.

University Virtual Learning Space provides as common to all students' services such as: e-Lecture; e-Test; curriculum timetable; scheduling of exam sessions and other events (e-Schedule), as well as specific services for different groups of students [9]. We will look at two specific services

for students with mobility disabilities:

- Ensuring a convenient route for moving the student's wheelchair to the study halls or labs;
- Providing of specific up-to-date information on changing the study hall and problems with the operation of automatic doors, elevators and ramps.

The student with movement difficulties will communicate with the system through his smart wheelchair and through his mobile device.

## CCA – Algebraic Semantics of Contexts

The Ambient plays a major role in Calculus of Context-aware Ambients (CCA) modeling [14]. We can treat it as a limited space in which certain actions are carried out. Each Ambient has a name, a boundary, and may contain other ambients in itself, and be included in another ambient. Between two Ambients there are three possible relationships: parent, child and sibling. Each Ambient can communicate with other Ambients around it by sending and receiving messages. The notation ":" is a symbol for sibling Ambients; "↑" and "↓" are symbols for parent and child; "<>" means sending, and "()" – receiving a message.

The Ambient can be mobile, i.e., to move within the surrounding environment. With CCA, there are two ways to move: in and out, which allow Ambients to move from one location to another.

In the CCA, four syntax categories can be distinguished: processes (P), capabilities (M), locations ( $\alpha$ ) and context expressions (k). Processes P are a basic syntax category in CCA modeling. Process 0 does nothing and ends immediately. The process  $P | Q$  indicates that the process P is performed in parallel with the Q process. The doubling sign is "!", so the process "!!P" is a process that can generate a new copy of P. The process  $n [P]$  denotes an ambient named n, and P is the process that describes the ambient behavior. The pair of brackets "[" and "]" predetermine the boundaries of the ambient. We can describe capabilities M from different point of view. Mobility options allow an ambient to move in its surroundings; the options to call a process are implemented as any actual parameter being replaced with its corresponding formal parameter; to send and receive messages is main to communication between ambients; to delete ambient can use "del n" (if it is empty).

## CCA Modeling of Intelligent Wheelchair

The wheelchair has a variety of physical sensors that collect information about changing environmental parameters and interact with the modeled virtual educational space. We will consider the wheelchair (and the active area that it creates) as a separate Ambient, with internal Ambients: the student (via his personal assistant PA<sub>i</sub>) and his mobile device for communicating with the learning environment. When the wheelchair physically enters in the university campus, an automatic identification process is started. After verification, the learning environment sends a response to the student's mobile device, containing a list of appropriate services for him. The

Ambients that we will use for modeling are: active wheelchair zone (CartZone) and the wheelchair (Cart), the student (Stu), the student's mobile device (Device), Active Zone of the Learning Environment (UniZone), Educational Portal (LP), learning resources (DiLib), administrative database (ADB), as well as important zones for the student and his wheelchair, such as lifts, ramps, doors, etc. (IZ) (figure 2).

When a wheelchair enters in the active campus zone, a guard assistant (GA) is activated to provide information on the activity and performance of the important zones for the wheelchair. Upon leaving the university campus, the active area of the wheelchair stops interacting with the learning space and follows the student's automatic logout, which terminates using of the provided services.

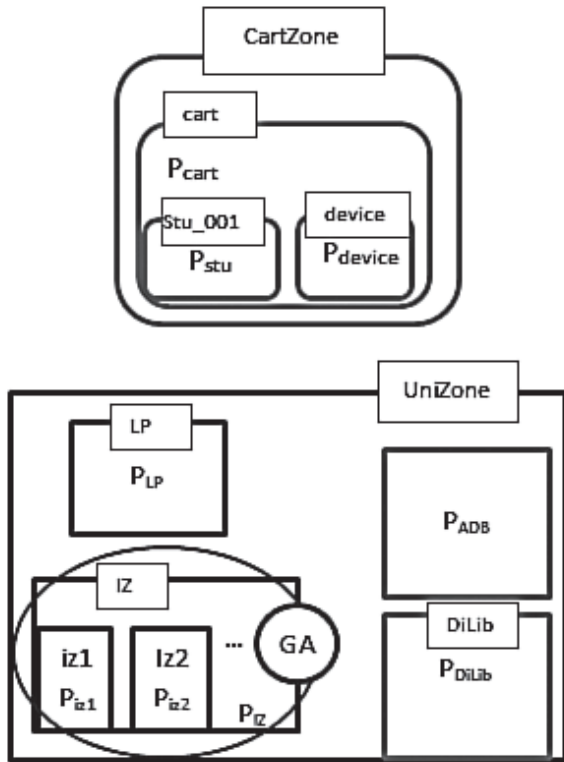


Figure 2. Ambients and their interaction

We can model through the following process:

CartZone[P<sub>zone</sub> |  
 car[P<sub>car</sub> | Student\_001[P<sub>stu</sub>]] | device[P<sub>device</sub>]] |  
 UniZone[P<sub>UniZone</sub> | LP[P<sub>LP</sub>]] | DiLib[P<sub>DiLib</sub>]] |  
 ADB[P<sub>ADB</sub>]] | IZ[P<sub>IZ</sub> | izID[P<sub>izID</sub>]]]

The student with mobility disabilities gets all learning services from the educational space. After recognizing the wheelchair (and the student in it), the system activates personal assistant (PAi) of the student, which takes care of the delivery of all the educational services and learning resources to his mobile device, like all other students.

The specific services that the environment will provide to this student are mainly related to his mobility in the physical environment of the university campus. Once the student receives the list of all services (sList) from his Personal Assistant (PAi), he understands at which floor and in which

room the relevant training session or exam will be held. The Guard assistant GA also receives this information from PAi and checks the functionality of all the important zones that the wheelchair needs to pass. It sends the current list to Device ambient. From the received information, the student chooses a route and sends it to the Cart ambient.

When the wheelchair moves in the physical campus, GA tracks its location and, when it is close to some of the wheelchair-critical zone, activates the sensors associated with opening the doors, providing a lift, etc. If in real time any of these important zones changes its status and becomes inactive, GA promptly informs the wheelchair (Cart Ambient), which expects the student to choose a new route. The route is a sequence of important zones (iz1, iz2 ...) We will describe the behavior of the main ambiens in the model.

**Ambient Cart.** As soon as the wheelchair enters the area of university campus, automatic login will take place and the ambient Cart becomes an internal ambient of UniZone, i.e. a sibling of LP, DiLib, ADB, and IZ. Similarly, in leaving the zone, an automatic logout follows:

$$(5.1) \left( \begin{array}{l} (\leftarrow at2(UniZone, cart)) ? \log in < cart > .0 | \\ !(\leftarrow \neg at2(UniZone, cart)) \\ ? \log out <> .(\leftarrow at2(UniZone, cart)) \\ ? \log in < cart > .0 \end{array} \right)$$

When the Cart ambient receives the list of active important zones from GA, it sends it to the Device ambient, from where it receives the route as a sequence of active important zones, i.e. route (iz1, iz2, iz5 ...). This route it sends to GA, along with information about its current location <location, route>, and starts moving according choosed rout.

$$(5.2) \left( \begin{array}{l} !GA :: (ListActiveIZ).Device :: < ListActiveIZ > .0 | \\ !Device :: (route(iz1, iz2, iz5...)). \\ GA :: < location, route > .0 \end{array} \right)$$

If Cart receives information from GA for dropping one of the important zones along the route due to failure or other reason, it restarts the process (5.2)

Consequently, the process of Cart ambient can describe as follows:

$$P_{cart} \triangleq \text{Eq.(5.1)} | \text{Eq. (5.2)}$$

**Ambient Device.** The student communicates with other ambiens through his mobile device, receiving both learning resources and services (like all other students). Furthermore he receives and information about the important zones needed to create a route for moving from his current location to the study hall. We will model the behavior of this ambient related to the wheelchair movement.

$$P_{Device} \triangleq \left( \begin{array}{l} !cart \uparrow (ListActiveIZ). \\ Stu\_001 :: < ListActiveIZ > .0 | \\ !Stu\_001 :: (route). \\ cart \uparrow < location, route > .0 \end{array} \right)$$

**Ambient GA.** This Ambient is internal to UniZone and communicates with other ambients – Cart and IZ. It activates when the wheelchair enters in the campus area, receives a message from IZ ambient (ListActiveIZ) whit information about the active important zones through which the wheelchair can pass and transmits this list to the Cart ambient. In response, it receives a message containing the actual location of the wheelchair and the sequence of important zones that must be passed during the route. Then GA sends the message (location, route) to IZ:

$$(5.3) \left( \begin{array}{l} !IZ :: (ListActiveIZ). \\ Cart :: < ListActiveIZ > .0 | \\ !Cart :: (location, route(iz1, iz2, iz5...)). \\ IZ :: < location, route > .0 \end{array} \right)$$

When the smart wheelchair approaches some important zone on the selected route, GA receives an IZ message for successful passage, or information that this important zone is no longer active, as well as an updated list of new active zones. In this case, GA passes the new list to the Cart ambient and the process is repeated.

$$(5.4) \left( \begin{array}{l} !IZ :: (\neg has(iz_i))?(reply, newListActiveIZ). \\ Cart :: < reply, newListActiveIZ > .0 | \\ !Cart :: (location, newRoute). \\ IZ :: < location, newRoute > .0 \end{array} \right)$$

Then:  $P_{GA} \triangleq Eq.(5.3)|Eq.(5.4)$

**Ambient IZ.** This ambient is responsible for providing information about the active important zones iz1, iz2 ... in the physical space of the university campus. He is a relative of Cart, GA, LP, ADB and ER and has internal ambients iz1, iz2 ... which we will group as izID. By means of sensors, it receives information about the activity of its internal ambients and provides it at the request of the other ambients in the model. After receiving a request from the GA containing the wheelchair location and the route selected by the student, IZ starts a process of dynamically calculating the distance from the wheelchair to the first important zone on the route. When this distance becomes less than a few meters, it activates the sensor that opens the door (or shouts the elevator, drops the ramp, etc. depending on type of the zone). If successful, the process is repeated for the next important zone on the route. If, in the meantime, this important zone has become inactive for technical or other reasons, IZ sends a message to GA with parameters (reply, newListActiveIZ), as "reply" is information about deactivation, and "newListActiveIZ" is the list of current active important zones. After receiving a message with the wheelchair location and the new route, it repeats the process.

$$P_{IZ} \triangleq \left( \begin{array}{l} !GA :: (location, route) izID \downarrow < location > .0 | \\ (!izID \downarrow ((has(izID)) and (is(distance < a)))) \\ ?izID \downarrow < start > .0 | \\ !izID \downarrow (\neg has(izID)). \\ GA :: < reply, newListActiveIZ > .0 \end{array} \right)$$

Modeling the services provided by VelSpace to students with mobility difficulties through ambient communication ensures distributivity and clarity at each step in the scenarios. This approach makes it possible to verify the interaction between the different components of the learning space and to correct inconsistencies and ineffective actions in the modeling and prototyping stage.

## Verification by ccaPL – Simulator

It is difficult to reproduce the syntax of CCA using a text editor due to the specific symbols that are included, such as "↑" and "↓". The language ccaPL is a computer-readable version of the CCA syntax. Tables 1, 2, 3 present the syntax of ccaPL in terms of processes (P), capabilities (M) and locations (α). Let h is a translation function from CCA to ccaPL, i.e. for the process P in CCA, h(P) is the corresponding ccaPL notation.

**Table 1.** Capabilities M in ccaPL

CCA: M	ccaPL: h(M)	Description
in n	in n	Input in ambient n
out	out	out
del n	del n	Delete ambient n
α (ý)	h(α)recv(ý)	Receive a list of messages ý from α
α <ý>	h(α)send(ż)	Send a list of messages ż to a α

**Table 2.** Processes P in ccaPL

CCA: P	ccaPL: h(P)	Description
0	0	Passivity
n[P]	n[h(P)]	Ambient with process
P Q	h(P) h(Q)	Parallel processes
!P	!h(P)	Duplication
k?M.P	<h(k)>h(M).h(P)	Contextually guaranteed precondition

**Table 3.** Locations α in ccaPL

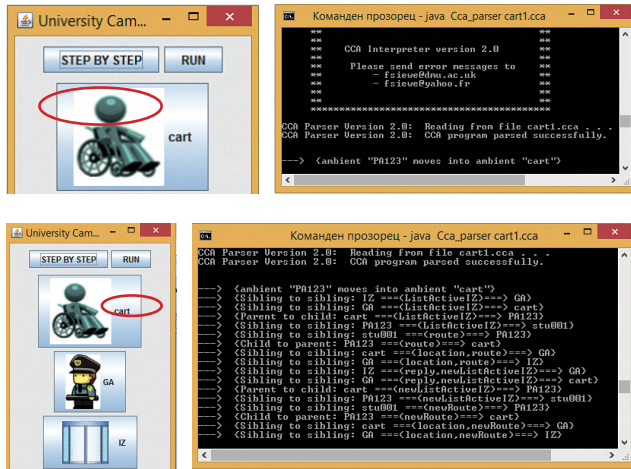
CCA: α	ccaPL: h(α)	Description
↑	@	Any parent
n ↑	n@	Parent n
↓	#	Any child
n ↓	n#	Child n
::	::	Any sibling
n::	n::	Sibling n

The interpreter of ccaPL has been developed as a Java application. Based on the main version [15], we developed a special program environment to verify the scenario described above. Each row of output data of the program represents the CCA relationships and interactions. The notation "A == (X) ==> B" means that Ambient "A" sends



an "X" message to Ambient "B". "Child to parent", "Parent to child," and "Sibling to sibling" provide information about the relationship between sender A and recipient B according to the hierarchy of ambients.

An animator has been developed to facilitate reading transitions [6]. The animator's goal is to present graphically the ambients and their processes. Transitions between processes are visualized; how and when is realized any opportunity for ambients; how every ambient moves in the surrounding environment. The animator presents the ambients hierarchically as the main one is presented as a container that contains all ambients-children. Each ambient-child is represented by a button with a name and an icon. The icons associated with the smart wheelchair model are: PA, GA, Cart, IZ, Stu. Internal ambients are also represented as buttons. The primary ambient container contains initially two buttons – START and STEP. When we run the program, the first transition is executed and the animation is displayed. The user then continues the scenario testing step by step or in full. Mobile transitions in the ccaPL-animator for this scenario are presented in *figure 3*.



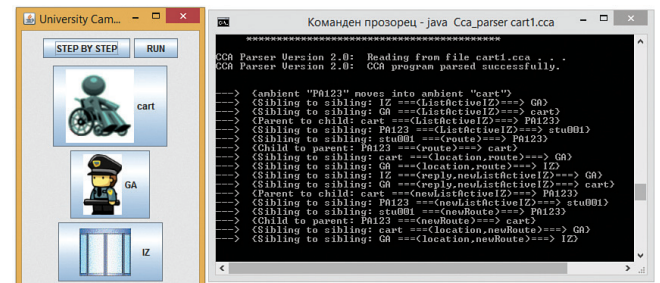
**Figure 3.** Visualization of transitions between ambients

The creation of a simulator and the validation of scenarios in the system are of particular importance for its realization. The purpose of this process is to prove that the system works correctly in accordance with the developed model. The scenario whose CCA modeling we presented has the following ccaPL program realization:

```
PA123[in cart.0]
cart[ GA::rcv(ListActiveIZ).
PA123#send(ListActiveIZ).0]
PA123#rcv(route).GA::send(location,route).0]
GA::rcv(reply,newListActiveIZ).
PA123#send(newListActiveIZ).0]
PA123#rcv(newRoute).
GA::send(location,newRoute).0]
PA123[cart@rcv(ListActiveIZ).
stu001::send(ListActiveIZ).0]
stu001::rcv(route).cart@send(route).0]
cart@rcv(newListActiveIZ).
stu001::send(newListActiveIZ).0]
```

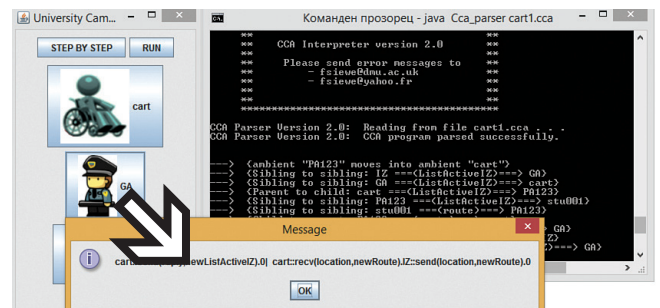
```
stu001::rcv(newRoute).
cart@send(newRoute).0 ]
stu001[
PA123::rcv(ListActiveIZ).PA123::send(route).0]
PA123::rcv(newListActiveIZ).
PA123::send(newRoute.0)]]
GA[IZ :: rcv(ListActiveIZ).
cart :: send(ListActiveIZ). 0]
cart::rcv(location,route).
IZ::send(location,route).0]
<not somewhere has(iz_i)>
IZ::rcv(reply,newListActiveIZ).
cart::send(reply,newListActiveIZ).0]
cart::rcv(location,newRoute).
IZ::send(location,newRoute.0)]
IZ[GA::rcv(location,route).
GA::send(reply,newListActiveIZ).0]
GA::rcv(location,newRoute.0]
GA::send(ListActiveIZ).0]
```

The implementation of the CCA-model in the ccaPL program environment (*figure 4*) allows to track the processes of the participating ambients as well and the sequence of the sent and received messages between them. The animator allows visualization of the participating ambients, their location and processes (*figure 4*).



**Figure 4.** Implementation of the scenario in the simulator

The processes of all participating ambients can be traced to each step of the scenario implementation, which makes it possible to immediately identify inconsistencies, errors and inaccuracies (*figure 5*).



**Figure 5.** Validation of processes and messages

Therefore the simulator in the ccaPL –program environment allows to track the sequence of the sent and received messages between the ambients according to the modeled scenario.

## Conclusion

The ambient-oriented CCA modeling of various scenarios related to the training of students with mobility problems enables the presentation and implementation of specific services that take into account the dynamic change in the environmental parameters. The programming language ccaPL allows describing the developed CCA model. The presented simulator provides a mechanism for verifying the correctness of the modeled scenario. The next step in the development process is verification of the model through its overall prototyping.

## Acknowledgements

The author expresses his gratitude to the scientific project FP17-FMI-008 "Innovative software tools and technologies with applications in research in mathematics, informatics and pedagogy of training" to the Fund "Scientific Researches" of Plovdiv University "Paisii Hilendarski" for partial financing of the present work.

## References

1. Schmidt, A. and K. V. Laerhoven. How to Build Smart Appliances. IEEE Personal Communications, ISSN 1070-9916, 2001, 66-71.
2. Dey, A. K. and G. D. Abowd. Towards a Better Understanding of Context and Context-awareness. Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness. New York, ACM Press, 2000.
3. Dey, A. K. Understanding and Using Context. – *Personal and Ubiquitous Computing Journal*, 5, 2001, No. 1, 4-7.
4. Glushkova, T. Adaptive Model for e-Learning in Secondary School. E-Learning, Book, Long-Distance and Lifelong Perspectives. ISBN 978-953-51-0250-2, 2011. <http://www.intechopen.com/download/get/type/pdfs/id/31951>.
5. Glushkova, T. Personalization and User Modeling in Adaptive E-Learning Systems for Schools, E-Learning – Instructional Design, Organizational Strategy and Management, Dr. Boyka Gradinarova (Ed.). ISBN: 978-953-51-2188-6, InTech, DOI: 10.5772/61084. Available from: <http://www.intechopen.com/books/e-learning-instructional-design-organizational-strategy-and-management/personalization-and-user-modeling-in-adaptive-e-learning-systems-for-schools>.
6. Mohammed H. Al-Sammarraie. Policy-based Approach for Context-aware Systems. PhD Thesis, Software Technology Research Laboratory, De Montfort University, Leicester – United Kingdom, July 2011.
7. Stoyanov, S., I. Ganchev, D. Mitev, V. Valkanov, M. O'Droma. Service-oriented and Agent-based Architecture Supporting Adaptable. Scenario-based and Context-aware Provision of Mobile e-Learning Services. – *International Journal of Computer Information Systems and Industrial Management Applications*, 3, USA, Dynamic Publishers, Inc., 2011, 771-779, ISSN 2150-7988.
8. Stoyanov, S., I. Ganchev, M. O'Droma, H. Zedan, D. Meere, V. Valkanova. Semantic Multi-agent mLearning System. A. Elci, M. T. Kone, M. A. Orgun (Eds.). *Semantic Agent Systems: Foundations and Applications*. Book Series: Studies in Computational Intelligence. Springer Verlag, 344, 2011, ISBN: 978-3-642-18307-2.
9. Valkanov, V., Context-oriented Management of Electronic Ser-

vices. Sofia, Bulgaria, Academic Publishing House "Prof. Marin Drinov", 2013, ISBN 978-954-322-701-3.

10. Kehayova, I., P. Malinov, S. Stoyanov. Intelligent Personal Assistants in a Virtual Learning Space, DeLC – Past, Present, Future. Proceedings of the International Conference – From DeLC to VELSPACE, Plovdiv, 26-28 March 2014, 29-40. Third Millennium Media Publications. ISBN: 0-9545660-2-5, <http://fmi-plovdiv.org/GetResource?id=1809>.

11. Stoyanov, S. Context-Aware and Adaptable eLearning Systems. Internal Report, Software Technology Research Laboratory, De Montfort University, Leicester, UK, August 2012.

12. Cardelli, L. and A. Gordon. Mobile Ambients. – *Theoretical Computer Science*, 240, 2000, 177–213.

13. Schwab Klaus. The Fourth Industrial Revolution. New York, Publ. Crown Business, 2017, ISBN 978-1-5247-5886-8.

14. Siewe, F., Hussein Zedan, Antonio Cau. The Calculus of Context-aware Ambients. – *Journal of Computer and System Sciences*, 77, July 2011, Issue 4, 2011, 597-620, doi:10.1016/j.jcss.2010.02.003.

15. Siewe, F., Antonio Cau, Hussein Zedan. CCA: aCalculus of Context-aware Ambients. Conference: Advanced Information Networking and Applications Workshops, WAINA '09. International Conference, 2009, DOI: 10.1109/WAINA.2009.23 · Source: IEEE Xplore.

## Manuscript received on 19.12.2017



**Todorka Glushkova** received M.Sc degree in Informatics, Plovdiv University "Paisii Hilendarski", Plovdiv in 2003 and PhD degree in Informatics and Computer sciences in 2011 from Plovdiv University too. Since 2013 she works in Faculty of Mathematics and Informatics in Plovdiv University as Assistant Professor. Her current research interests are in Modeling of Context-aware Educational Spaces, Adaptive e-Learning services and e-Learning resources.

### Contacts:

Plovdiv University "Paisii Hilendarski"  
236 Bulgaria St., 4000 Plovdiv  
tel: +359 878212950  
e-mail: [glushkova@uni-plovdiv.bg](mailto:glushkova@uni-plovdiv.bg)