DEVELOPMENT OF A COMPONENT FOR CONTEXT-AWARE MODELING IN VIRTUAL-PHYSICAL SPACE

Konstantin Rusev, Todorka Glushkova

Abstract. *The paper presents one approach for developing a prototype of a Component for CCA modeling in Virtual-Physical Space (ViPS). The results of the first prototype version of this Component are presented, in which the spatial aspects of "things" are mainly modeled. A simple demonstration example is reviewed and directions for future development are commented.*

Key words: Virtual-Physical Space (ViPS), Calculus of Context-aware Ambients (CCA).

Introduction

The Fourth industrial revolution (Industry 4.0) shares the understanding that the enormous potential of artificial intelligence, virtual reality, the connection between the physical and virtual worlds, etc. could be used to improve the quality of life of the people in the modern world. The three closely related paradigms of the Internet of Things (IoT), Cyber-Physical Systems (CPS), and Cyber-Physical-Social Systems (CPSS) are considered to take a key role [1]. Due to the high level of complexity, the creation of such systems requires extensive preliminary research, modeling, and testing of the main interactions, scenarios, and services.

According to the National Science Foundation (NSF) [2] definition, Cyber-Physical Systems (CPS) are designed to ensure the seamless integration of computational algorithms and physical components. Cyber-physical-social system (CPSS) is a fusion of physical space, cyberspace, and social space. In recent years, a reference architecture of Virtual Physical Space (ViPS) has been developed at the Faculty of Mathematics and Informatics of Plovdiv University [3]. In its essence, it is a CPSS space architecture that could be adapted to different application areas – e-learning [4, 5] agriculture [6], tourism, smart city, etc.

There are different definitions of the context. We base our research on Dey's definition [7], where context is any information that could be used to categorize the state of identity. Identity could be a person, place, or object which is believed to be associated with the interaction between a user and an application, including the user and the application itself. The formalization of context-aware systems is essential for their representation, design, and development. There are different formal methods such as Model-based methods; Algebraic methods; Methods based on Process Calculus; Logical methods; Network-based methods, etc.

Cyber-physical systems are dependent on both the surrounding world and the changing of the current context. Due to the complexity of their development, they require a long process of preliminary modeling, testing and verification of the main functionalities, processes, scenarios and services. The publication presents an approach to the development and testing of a modeling component that is based on Process Calculus methods and in particular on Ambient-oriented modeling.

Component for CCA modelling

The Virtual Physical Space (ViPS) is being built as a reference architecture that could be adapted to various CPSS applications in different application areas (domains). The ViPS architecture provides the virtualization of real objects that are related to the considered domain. The virtualization of the "things" is supported by the ViPS middleware, which includes two main subspaces – the "Analytic Subspace" and the "Digital Libraries Subspace", as well as the "Event Manager" [8] and operational assistants with different functionalities and responsibilities. The Analytical subspace provides tools for digitizing and representing information related to time, space, and location of objects. In this sense, the analytic subspace is built from the following basic components:

- TNet (Temporal Net) provides an opportunity to represent the temporal aspects of the "things", events and locations;

- AmbiNet (Ambient Net) provides the ability to model the spatial characteristics of "things" and events through context-aware ambients.

The work with these components is supported by specialized interpreters implemented in ViPS, based on the official specifications of Interval Temporal Logics (ITL) [9] and Calculus of Context-aware Ambients (CCA)

[10].

Therefore, the modeling of the processes and evaluating individual plans in the Analytical Subspace requires the creation of a Component for CCA Modeling into the scope of AmbiNet. In this report will present the first version of this Component in which the spatial aspects of the objects in the process of their interaction are considered.

According to the accepted definition, the base context is represented as the ordered four items: (*identity*, *location*, *activity*, *time*). That is:

$$k = (identity,\ location,\ activity,\ time). \tag{1.1}$$

Each change in these categories changes the context too. Consider the multitude of different contexts in the following scenario $K = \{k_1,\ k_2,\ k_3,\ \ldots,\ k_n\}$, as $k_i \in K$ and $k_i = (ID_i,\ LOC_i,\ ACT_i,\ TIME_i)$, for each $i = 1,\ \ldots,\ n$.

In real time, the Guards monitor changes in environmental parameters, which automatically causes changes in the current context. In modeling mode, it is necessary to simulate the change of the environment parameters, which will lead to a change in the context.

AmbiNet works in modeling mode. The ambients formalize identities involved in the test scenario. Since they unify the processing of objects from the physical and virtual worlds of the CPSS space, it is necessary to expand their characteristics. By definition, Ambients represent separate identities and their locations in relation to the Ambient hierarchy. Based on the hierarchy each ambient has only one parent. The location in the hierarchy is uniquely described by the parent ambient (if there is no parent, we write "None"). Because the Ambients are mobile and can change their location in the Ambient hierarchy, this structure changes dynamically. Furthermore, each Ambient has a defined set of processes "$P(a)$". Then, we can consider the Ambient as the ordered triple items:

$$a = (Name,\ Parent,\ P(a)). \tag{1.2}$$

In addition, it is necessary to indicate whether the identity represented by the Ambient is from the physical or the virtual world. For physical identities, it matters whether or not they can change their location in the real world. For example, the Museum represents an object from the physical world which is static. The static physical Ambients also have a specific geographic location. Then we can expand the presentation of Ambient with a few more characteristics. Thus the Ambient could be represented in AmbiNet as an ordered list (*identity*, *parent*, *type*, *mobility*, *location*, *processes*):

$$a = (Name, Parent, Type, Dynamicity, Location, P(a)). \tag{1.3}$$

The set of ambients represents identities. In the sample scenario this could be $A = \{a_1, a_2, a_3, \ldots, a_m\}$, where $a_i = (ID_i, PAR_i, TYPE_i, DYNAM_i, LOC_i, P(a_i))$.

In CCA, two ambients can exchange messages if they are in one of the following relationships: "parent", "child", or "sibling". Then, the ambients "a_i" and "a_j" can communicate directly if they are in one of these relationships. Otherwise, they pass the messages through the ambient hierarchy.

$$(a = parent(b))OR(b = parent(a))OR(parent(a) = parent(b)). \tag{1.4}$$

Each change of the context can lead to a change in the ambient hierarchy, or in the interaction between the ambients in the modeled scenario. Therefore, in a simulation mode, the change of contexts is reflected in the development of individual base scenarios for the corresponding application area. These scenarios can be formalized through the relationships between the CCA ambients which are represented in the conceptual model.

Due to the specifics and goals that it has to fulfill, the developed component is part of the AmbiNet of the Analytical Subspace. Based on the expected change of context information in the modeled application area, the example base scenarios describe the processes of realization of the expected functionalities of the identities. For the CCA-based modeling of these scenarios, an AmbiNet CCA Editor is developed to enable a visual, and largely intuitive workflow which is consistent with the model presented above. While this editor provides additional advanced functionality, it must support the requirements of the CCA formalism. Therefore, the generated models must be able to be executed, tested, and verified by the classic standardized CCA-Interpreter [11]. To provide the ability to work in integrated domains, as well as to use and manage the additional characteristics of ambients, it is necessary to create a Data module. It will store both the current information about the ambients and the application domains in the context of these ambients. In addition, operational information about the CCA editor's work such as cca files, CSV data representations, messages, etc. could be stored in this module. The development of a 3D simulator will provide a clear visualization of the processes in the modeled scenario. The generated and tested scenarios are stored in the AmbiNet Repository in a form suitable for future use.

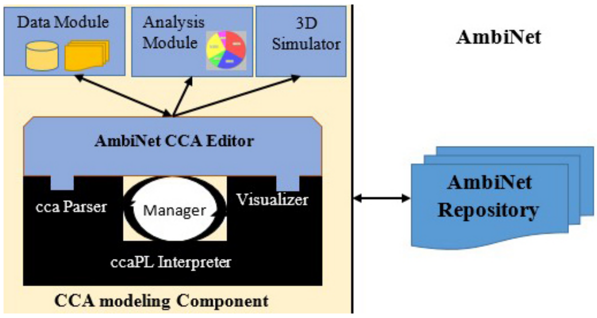

Figure 1. Structure of a component for CCA modeling.

Based on the presented model, a software application is being developed to support and facilitate the process of developing CCA models. In the current version, the application implements the process of creating, storing, and reusing scenarios to model processes in integrated domains. The additional capabilities for modeling in integrated domains and analyzing the interactions between the identities (ambients) provide opportunities for pre-adjustment and optimization of the modeled scenarios and processes. The Modeling Component embeds the standardized classic ccaPL Interpreter, which ensures the correctness of the models in terms of the formal CCA semantics.

To test and verify the operations of the Component we use base scenarios developed manually. The results obtained during testing confirm and verify the correctness of the developed Component. Let's consider the following sample scenario: *A student, through their personal assistant (PA), communicates with the educational platform of Plovdiv University "Paisii Hilendarski", and with other units (ambients) that are part of it (seminar hall, library, and others) in order to obtain the necessary information and to allow the student to participate the upcoming lectures. The student receives a bunch of materials such as lectures, books, textbooks, articles, and other resources.*

The modeling of the scenario is done through the developed CCA editor interacting with the other modules of the Modeling Component. Initially, the modeler must select an application domain in terms of which the CCA scenario will be implemented. In this case, a model will be implemented in the applied area of "e-learning" (Fig. 2). Then, it should create (or use previously recorded in the Data module) ambients processed according to the semantics presented above such as:

- $a_1(StudentPhone, University, Abstract, None, None, P(a_1))$;
- $a_2(University, Plovdiv, Physical, Static, loc\_University, P(a_2))$;
- $a_3(Library, University, Physical, Static, loc\_University, P(a_3))$, etc.

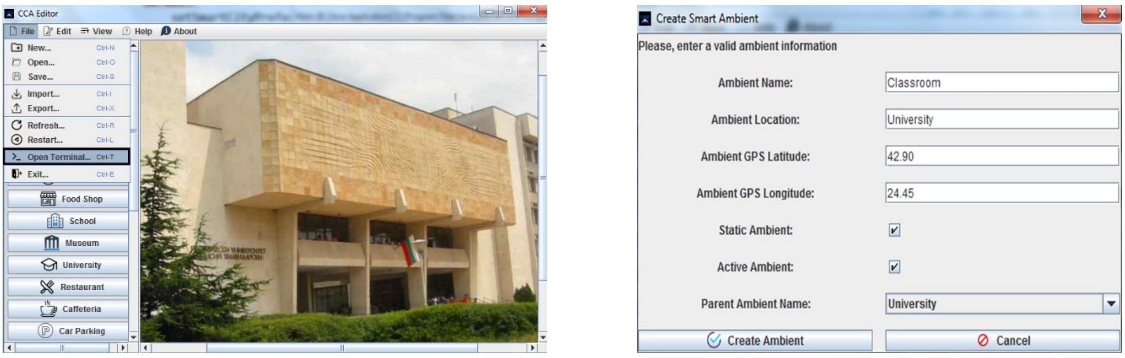

Figure 2. Create ambients in educational domain.

As the next step in the implementation of the scenario, communication between the already created ambients must be carried out. This happens by creating messages. The next stage of modeling is model generation. During this process, all ambients and messages are accessed, processed, and converted using a special programming language (ccaPL). The final result is saved in a file which is stored in a directory designated for the generated CCA models (Fig. 3).

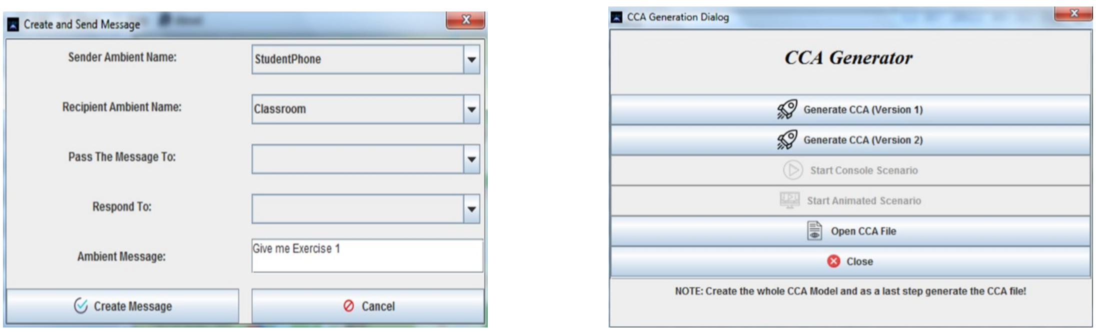

Figure 3. Create messages and generate CCA model.

Once the CCA file is generated, it could be executed using the classic "ccaPL" programming language interpreter. Before the execution, the generated file could be visualized and edited in the CCA editor. Once the CCA file is generated and the modeler has reviewed and/or edited it, it can be moved to the final step of the scenario implementation – running, testing, and verifying the CCA model using the "ccaPL" programming language interpreter (Fig. 4).

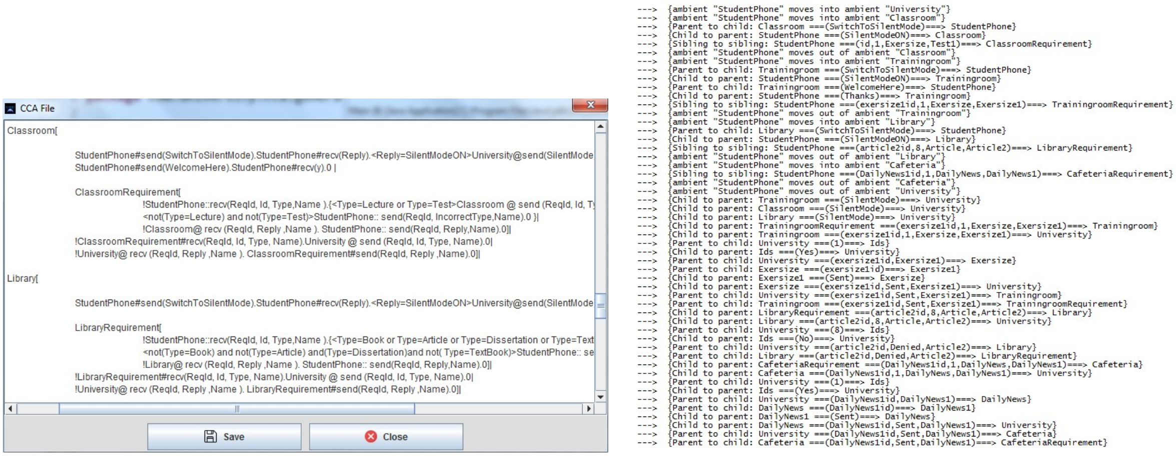

Figure 4. Generated CCA model and results of execution in the ccaPL interpreter.

The last step is an analysis of the achieved results. This is possible with the help of the Analysis Module. It is developed to provide a set of ambient and message statistics. The purpose of this model is to optimize the created scenario (Fig. 5).

The already generated CCA models are stored in the Data Module and could be used in subsequent modeling and optimization processes.

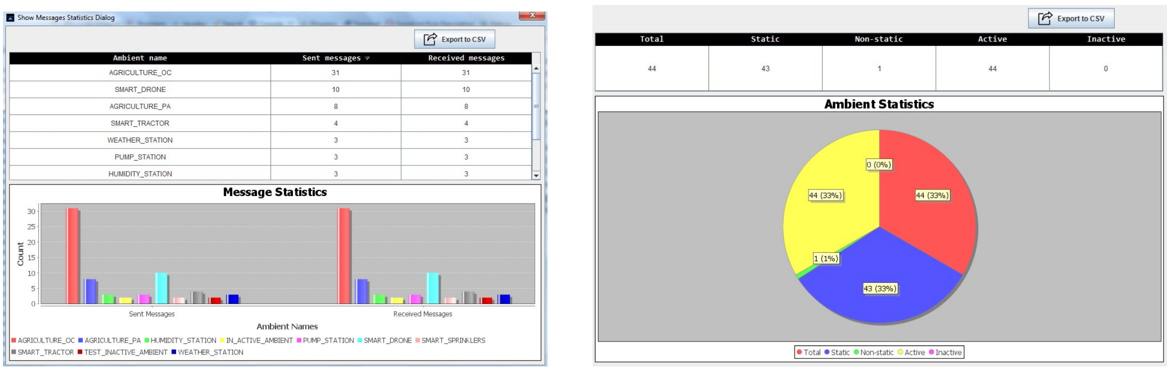

Figure 5. Statistics provided by the Data Module.

Conclusion

Building a Virtual Physical Space (ViPS) is a complex, time-consuming, and financially risky task. The presented CCA modeling Component prototype enables developers to create and test the designed scenarios in the pre-development stage in a visual and intuitive environment.

The use of context expressions in CCA modeling enables the "capturing" of the context and the implementation of certain possibilities related to the behavior of the BDI agents involved in the modeled processes. Future plans are to continue work on the considered Component, which will allow modeling of processes not only in terms of spatial aspects, but also in the conditions of a dynamically changing context-aware environment.

Acknowledgments

The results published in this article are part of a study conducted with the financial support of project FP21-FMI-002 "Intelligent Innovative ICT in Research in Mathematics, Informatics and Pedagogy of Education" of the Scientific Fund of Plovdiv University "Paisii Hilendarski" in Bulgaria.

References

[1] S. Stoyanov, T. Glushkova, E. Doychev, *Cyber-Physical-Social Systems and Applications – Part 1*, LAP LAMBERT Academic Publishing, 2019, ISBN: 978-620-0-31825-1.

[2] National Science Foundation (US), `https://www.nsf.gov/pubs/2008/nsf08611/nsf08611.htm`, visited on 31.07.2022.

[3] S. Stoyanov, A. Stoyanova-Doycheva, T. Glushkova, E. Doychev, Virtual Physical Space – An Architecture Supporting Internet of Things Applications, *20th International Symposium on Electrical Apparatus and Technologies (SIELA)*, 3–6 June 2018, `DOI:10.1109/SIELA.2018.8447156`.

[4] J. Todorov, I. Krasteva, V. Ivanova, E. Doychev, BLISS-A CPSS-like Application for Lifelong Learning, *IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, 2019, 1–5, `doi:10.1109/INISTA.2019.8778363`.

[5] M. Grancharova-Hristova, N. Moraliyska, S. Madanska, Development of an ontology in the field of the humanities, Proc. of

International conference "CULTURAL AND HISTORICAL HERITAGE Preservation, Presentation, Digitalization – KIN21", Veliko Tarnovo, March 2021, `http://www.math.bas.bg/vt/isc-kin/files/KIN2021-book-abs.pdf`.

[6] A. Stoyanova-Doycheva, V. Ivanova, L. Doukovska, V. Tabakova-Komsalova, I. Radeva, S. Danailova, Architecture of a Knowledge Base in Smart Crop Production, Proc. of *International Conference Automatics and Informatics (ICAI)*, 2021, 305–309, `doi:10.1109/ICAI52893.2021.9639874`.

[7] A. Dey, Understanding and Using Context, *Personal and Ubiquitous Computing Journal*, Vol. 5, No. 1, 2001, 4–7.

[8] S. Stoyanov, A. Stoyanova-Doycheva, V. Ivanova, V. Tabakova-Komsalova, V. Monov, Z. Radeva, An Event Model for Smart Agriculture, Proc. of *International Conference Automatics and Informatics (ICAI)*, IEEE, 2021, 314–317.

[9] V. Valkanov, A. Stojanova-Doycheva, E. Doychev, S. Stojanov, I. Popchev, I. Radeva, AjTempura – First Software Prototype of C3A Model, *IEEE Conf. on Intelligent Systems*, 2014.

[10] F. Siewe, A. Cau, H. Zedan, CCA: a Calculus of Context-Aware Ambients, Proc. of International Conference *Advanced Information Networking and Applications Workshops* WAINA'09, 2009, `DOI:10.1109/WAINA.2009.23`.

[11] F. Siewe, ccaPL: a Programming Language for the Calculus of Context-aware Ambients, *Software Technology Research Laboratory*, Leicester: De Montfort University, 77 (4), 2011, 1–11.

Konstantin Rusev[1], Todorka Glushkova[2,*]
[1,2] "Paisii Hilendarski" University of Plovdiv,
Faculty of Mathematics and Informatics,
236 Bulgaria Blvd., 4003 Plovdiv, Bulgaria
Corresponding author: `glushkova@uni-plovdiv.bg`