U.P.B. Sci. Bull., Series C, Vol. 81, Iss. 2, 2019 ISSN 2286-3540

INTEROPERABILITY ASSESSMENT IN CONTEXT-AWARE SYSTEMS FOR INTELLIGENT TRANSPORT MANAGEMENT

Celestin DRAGANESCU[1], Aurelian STANESCU[2], Radu DOBRESCU[3]

This paper aims to reveal evidence on how interoperability has been addressed in context-aware systems, presenting a set of characteristics that can support decision-making and evaluation of interoperability on such systems. Assuming that model-based interoperability refers to a collaboration of independently developed systems, we propose a hybrid approach to model contextual information that incorporates the management and communication benefits of traditional object-oriented context models. A model of such type has been designed to access and correlate distributed knowledge in the transportation domain. The context-aware model is part of a middleware architecture for integrating independent Intelligent Transport Systems (ITS) and pervasive transportation services. Using a distributed, tiered approach, this framework facilitates the sharing and collating of data. The performance evaluation was realized by analyzing data related to benefits, such as the time it takes to retrieve live data and the time it takes to reason over this data dynamically. Finally, in a Case Study, a solution for implementing Location-Based Services using model-based interoperability is presented.

Keywords: interoperability, context-awareness, Intelligent Transport Systems, pervasive middleware

1. Introduction

Rapid expansion of pervasive and ubiquitous computing, logistically benefiting from the evolved support of informational networks (Cloud, Fog, IoT) leads to embedding of computer systems into everyday life environments, first in the intimacy of smart houses, then very soon, outwardly in the complex networks of Intelligent Transportation Systems (ITS). The common goal of such systems is to enrich the user experience without demanding his/her explicit attention. For pervasive computing to be minimally invasive, computer systems must be conscious, or, conform to a more used term, context-aware. Different software and

[1] PhD student, Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest, Romania, e-mail: celestin.draganescu@gmail.com

[2] Prof., Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest, Romania, e-mail: amstanescu@yahoo.com

[3] Prof., Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest, Romania, e-mail: rd_dobrescu@yahoo.com

hardware technologies can be integrated and used to capture context information, without user intervention. However, the heterogeneity of technology makes interoperability a great development challenge. The context is often heterogeneous as it is derived from a huge number of independent systems and sensors, and so it can be incomplete, and may even be erroneous. For independent systems to manage, share, correlate, and reason over context, contextual information must be modeled in a homogenous fashion. The interoperability is called to play this role. Interoperability implies the interaction of heterogeneous sets of embedded systems which are developed by independent producers. The true challenge there is how to achieve a sustainable, open, and autonomic interoperability, robust for independent system development and independent interpretation of interoperability standards.

More and more often, interoperability is connected with another feature of informational systems, the context-awareness, employed due to their usefulness to enhance systems functional reliability. A system can be termed as context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task. A context-aware system (CAS) adapts and provides relevant information and the most appropriate service to users in an active and autonomous manner while requires little interactions. In this paper we consider these issues and the challenges of model-based interoperability for a CAS, with insights from a case study in the ITS domain. The paper is organized as follows: First some papers having similar objectives are discussed. Then is presented a selection of interoperability characteristics, influenced by context-awareness. Subsequently, we present the aspects of a model-based interoperability framework for contextual information management and a pervasive middleware architecture for integrating independent ITSs. All these issues are considered to solve a transportation problem in a case study. The paper ends with a discussion on the results and options for future research.

2. Related Works

This work is related to three significant thematic areas: Context Awareness (CA), Interoperability (I) and Intelligent Transportation Systems (ITS). For each of these, the literature is very rich. We have also found numerous papers about how I and CA relate, as well as to both CA and ITS, but very little for the way I is reflected in the ITS area. Instead we did not find (at least with the means of documentation we have), no work to address all three topics, perhaps with the exception of an older work [1] in which for a context-aware system dedicated to the dissemination of Travel Route Information is mentioned the interoperability between data sets captured across individual distributed systems through its hybrid nature. This could be a good sign for the originality of our work, but we are pleased to show only that it is part of the tendency to extend the role of interoperability in as many applications as possible. In the following, we will refer to some papers in which we have found common issues with those addressed in our work, trying to compare the performance of our proposed solutions with the achievements mentioned in the cited references.

Among the works that discuss the importance of interoperability in context-aware systems, we first selected two works by the same group of authors, Motta et al ([2], [3]), both aiming at revealing evidence on how interoperability has been addressed in context-aware software systems. As in our work, they present a set of characteristics that can support evaluation of interoperability on such systems, but don't take into account the context in a natural environment like ITS. In [4] is proposed a context management system as component of a context-aware middleware that supports distributed context-aware applications to obtain context information from pervasive computing environments. Like in our work, communication interoperability is addressed, as essential function to adapt the system to context dispatching, fault tolerance, and network authentication. Again, the paper refers only to context-aware software systems.

Although, as we already mentioned, there are many works on context-aware ITS, we have retained only three that address similar issues with those discussed in our paper: paper [5] - for Multi-modal traffic management in ITS; paper [6] - for the idea to use Cloud-based architecture for context-aware ITS and paper [7] - for the connection between ITS and cyber-physical systems (CPS). In [5] is introduced a collaborative model-based context awareness multi-modal traffic management aiming at providing an efficient way to manage the traffic inside a transportation station, using a Q-learning technique. In [6] is presented a Cloud-based context-aware service framework which enables data processing in a specific application context and facilitate application layer to correlate information based on end users' preferences. Paper [7] proposes a new context-aware based modeling method for the intelligent transportation cyber-physical systems (IT-CPS), which controls the derivative process based on scene information inference.

To highlight how interoperability has been used in ITS we have selected two references. Paper [8] studies the impact of differences in the choice and construction of abstractions of a real-world entity to fit a specified model framework (defined as abstraction conflicts) on model-based interoperability in ITS. Paper [9] is a solid research report especially interesting for how to apply the interoperability standards in the geographical area to which Romania belongs.

Finally, we mentioned two papers ([10], [11]), both addressing the real-time data exchange between vehicles with wireless communication. They propose a distributed information interaction system for the interoperability between various devices, issue which is addressed in the case study presented in this paper.

3. Addressing interoperability in context-aware systems

Context-awareness becomes an important feature when considering the particularities of this kind of technology. A context-aware system (CAS) can perceive, understand and use contextual information to self-adjust to the current situation. Different technologies such as devices (sensors, computers, mobile technologies), applications and services should work together to capture and deal with context information. As a consequence, interoperability becomes essential since different types of computing technologies must interact. We considered in the following the simplest interoperability definition, namely „the degree on which two or more systems, products or components can exchange information and use such information" [12]. We studied then the possible context-awareness influence on the interoperability complexity, with the purpose of finding interoperability characteristics in the context of CAS.

At the first hand we founded that interoperability can be interchanged with connectivity, integration or communication, in all situations in cooperative systems. After setting the general interoperability perspective, one of the challenges in investigating CAS was exactly the term context. From the context-awareness perspective context can be considered an interoperability dimension. It refers to any information acquired by a sensor where the results can differ depending on different context information. The challenges to achieve true interoperability due to context can be associated to: a) heterogeneity of networks; b) different connectivity requirements in several forms (smart things, mobile technologies); c) poor application portability and increasing platform dependency and structural challenges.

Our opinion is that interoperability cannot be directly observed but perceived through its related characteristics. Table 1 resume the main interoperability characteristics associated with CAS.

Table 1

Interoperability Characteristics.

Characteristic	Description
Conformance with requirements	Related to the need to satisfy user requirements
Share common concepts	Understanding concepts, terms, and processes relevant to the interacting systems
Architecture definition	Architecture approach and style, platforms and other properties
Data definition	Concerns about data type, structure, representation and format
Protocols definition	Select the suitable protocol for each interaction level
Standards definition	Select the suitable standard for each interaction level
Use of services	Discover the specific service which offer a

	solution
Use of transformation	When some modification is required for the systems to interact
Use of Interfaces	Related to the identification and interface conformance
Dynamic connection	Authorize identified actors to connect according to permissions
Adaptive behavior	Ability to adapt dynamically to the environment
Systems Availability	Related to connectivity: during the interaction all systems should be available
Planning	Decisions made at design-time or run-time

When taking context-awareness feature into account, we retrieved information like monitoring changes of state to trigger actions within the environment; dynamically modify their information processing approaches and representations; runtime monitoring to maintain information up to date, and monitoring service behavior. All of these consider context variance and modifications at runtime to perform according to the changes. These characteristics are related to particularities of CAS requiring an adaptive behavior. This behavior refers to the system ability to adapt itself dynamically to the environment where it is used within its limitations, calling for monitoring the systems environment and adjusting it to a different condition.

4. A model-based interoperability framework for contextual information management

Context-aware Systems (CAS) adapt and provide information or services obtained from the external environment to the controlled process. In this paper the aim was to study the potential application of CAS for supervision and management of intelligent transport systems and definition of an architecture dedicated to a Vehicle Context-aware System (VCAS). VCAS is a system that must fulfill several tasks:

1) Collect, analyze and manage context information relating to the vehicle, the driver and the environment in which the vehicle travels. In particular it is desirable that the context information to fit in a predetermined pattern.

2) React through feedback to changing context and the changing requirements of the driver.

3) To follow the relationship between changes in context and systemic reactions. Depending on the effect on the system, context information can be classified into two categories: operational context, used to keep the system operating within normal parameters and management context used in the process of change/adapt of the parameters.

As such, the surveillance provided by the VCAS includes planning system monitoring, detection and diagnosis of anomalies and predict possible traffic events and the management service ensure effective action to ensure system reliability and availability. The software architecture of the system, VCAS is to be considered as a middleware between primary context data acquisition and the high-level context information management (Fig.1).

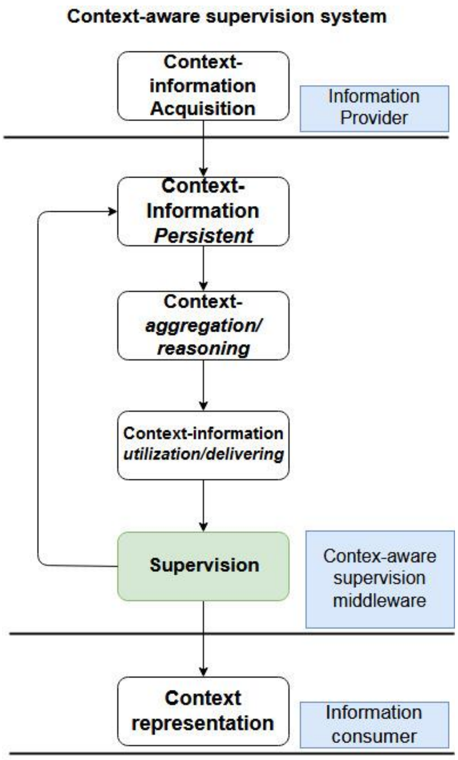

Fig. 1. VCAAS abstract architecture

The supervision block is the central element of the middleware VCAS level. It is important to note that the information flow between Level 1 (low) and Level 3 (upper), which is basically unidirectional, has a feedback loop which allows data processing after the context model. Doing so not only raw data are context-aware, but data from the monitoring process will be aggregated with other relevant information to bring relevant information to the end user. Moreover, surveillance information may serve to deduce new knowledge and propose new ones.

5. A pervasive middleware architecture for integrating independent ITSs.

The transportation domain is a canonical example of an area that can benefit greatly from pervasive middleware architectures. An abundance of context information is continuously collected, processed, and stored by ITS. However, the context produced by ITS is mainly system specific and therefore cannot be easily shared with or used by other systems. We propose a dedicated framework for Context-Aware Intelligent Transport (CAIT) management which is a middleware architecture for integrating ITS in large scale pervasive environments. Using a distributed, tiered approach, it facilitates the sharing and collating of data, while allowing the scalable integration of new and legacy ITS

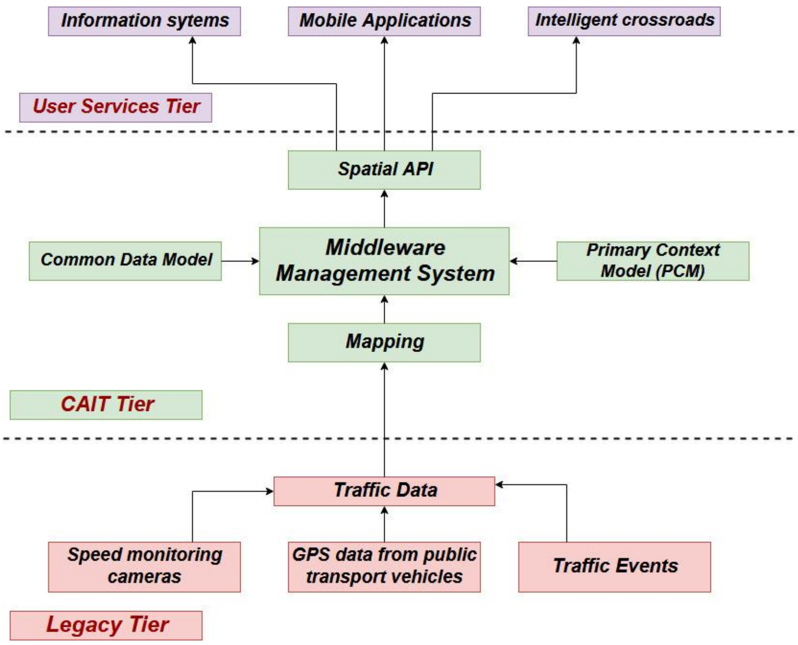

Fig. 2. Overview of the CAIT

As illustrated in Figure 2, the CAIT framework structures legacy systems, ITS, and context-aware, end-user applications and services into three tiers. The legacy tier provides for the integration of existing systems that were not developed to conform to the CAIT system architecture, but which still provide valuable information that can be used by pervasive transportation services. The CAIT tier comprises a federation of ITS that have adopted the CAIT system architecture, implying that they model their context according to the Primary Context Model (PCM). In our framework, these ITS are known as CAIT systems. The common data model is a key component of the architecture, as it represents all context available in the domain (in particular a VCAS system). It facilitates the integration of context from existing CAIT systems that were designed specifically as part of the CAIT framework, as well as those CAIT systems that act as wrappers for legacy systems. CAIT systems use spatial objects to model their context and implement the Spatial API, to provide access to these objects. Each system models the subset of the spatial objects that is relevant to its respective purpose and context-aware user services exploit the Spatial API to integrate and share information in a common way regardless of the specifics of the system implementing a particular part of the data model.

The application tier includes the pervasive services that provide context-aware user access to and interaction with traffic information. These pervasive services use the distributed data model and the associated context to access information potentially provided by multiple systems and might include a wide range of interactive (Internet-based) and embedded control services ranging from monitoring of live and historical traffic information to the display of road network maps.

Context data varies in its level of importance with regard to the interaction between systems or between systems and users. Primary context (PC), however, is the most significant context data, as it is used as the basis for querying other context, and it can therefore be used to correlate context from different systems. PC is identified in different forms as identity PC, location PC, time PC, and quality of service PC. The PCM provides a standard way for systems to store, manage, and share context in a scalable manner, based on primary context. Such a thorough representation of all knowledge in a domain can be used to correlate data from different sources and, ultimately, to reason about the context and to infer new context.

6. Case study

Our study has as goal to use interoperability in order to avoid conflicts between information acquired from different digital maps. More precisely, we want to correct errors due to the semantic heterogeneity, i.e. differences in the meaning of the information contained in the model. Additionally, we try to prevent also errors due to the structural heterogeneity (differences in the collection of real-world entities modeled)

Figure 3 shows one real-world example of model heterogeneity between maps of different vendors. The map on the left may not have a side street which is actually present in the other map (structural heterogeneity). A road might have a different geometry or topology in the map on the right (semantic heterogeneity).

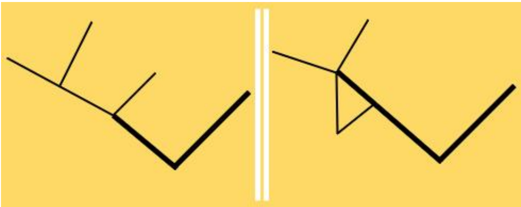

Fig. 3. Differences between digital maps

We have three main categories of conflicts due to map heterogeneity: 1) Naming conflicts (names of schemes and conventions differ significantly); 2) Scaling conflicts (different reference systems are used to measure values); 3) Semantic conflicts (information items appear to have the same meaning but differ in reality due to a different temporal or spatial context).
In the ITS domain, Location Based Services (LBS) provides important services as traffic telematics and tourist information (e.g. hotels, restaurants, monuments).

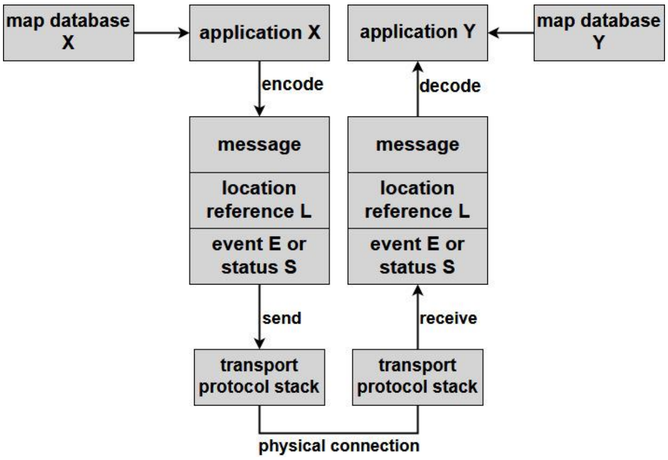

Fig. 4. The role of a location reference in a LBS application

The typical service chain for a LBS is shown in figure 4. An application X (e.g. at a traffic telematics service provider) constructs a message about an event E or status S (e.g. a traffic problem) on a location R. The message thus contains a location reference L, derived from representation of location R in the digital map X. This message is transmitted via a protocol stack to be received by application Y, using a different digital map Y to reconstruct location R from the received location reference L. This location reference L is thus the key customizing element. Location referencing demands model-based interoperability to be successful. Two basic location referencing methods can be distinguished:

• Pre-coding of location references: Pre-agreed location codes are listed in a location table for inclusion in digital maps.

• On-the-fly (or dynamic) coding: the on-demand creation of temporary location references from a model of the location in the digital map.

Pre-coding has as advantage a very compact code size for transmission but implies large maintenance and administration costs though, and therefore is only used to code relatively few, important locations on the major parts of the road network. On-the-fly model-based location coding does not have the disadvantages of the pre-coding method. Most important, the whole road network is addressable. This makes on-the-fly location referencing attractive as model-based interoperability mechanism, but only if it can overcome the issue of semantic heterogeneity between digital maps.

Our method for on-the-fly location referencing explicitly acknowledges the significant presence of model heterogeneity in digital maps. From the onset on, the reconstruction of a location on a second map was considered as a 'matching' problem. This method employs not just one single matching technique, but rather three complementary options: (1) shape geometry, (2) road section typing (form-of-way, road descriptor, functional class, and driving direction) and (3) topological connectivity. The redundancy in matching options turned out to be very useful. A source of information could be used either for matching and obtaining candidate locations (to overcome semantic heterogeneity), or for confirmation or rejection of candidate locations (to overcome intentional heterogeneity).

In the end, the method consists of the following elements:

1. The encoding framework including the building blocks and a list of allowed attributes;

2. The logical data model (a flexible set of elements that succinctly model the location to be coded);

3. The prescribing lower bound (i.e. minimum) requirements on a location reference constructed by an encoder;

4. A detailed description of the reference physical format;

5. Guidelines for an encoding procedure.

The encoding procedure is based on the encoding rules and may be used as a guideline to develop an encoding algorithm. In addition to the defined highly efficient bit-oriented physical format for in-vehicle applications, other physical formats may be defined, as long as these represent the defined encoding rules.

A key objective in the method is to permit an encoder flexibility in the choice and location of "anchoring" points, that is, the points were first contact is made with the second map. Prior methods required such points to be exactly located on a possibly complex intersection: locations where many road sections cross or run in parallel, and where much confusion can arise.

This case study has been selected because maps are one of the fundamental elements of ITS. This method of handling differences between database components could be extrapolated to other types of ITS elements (route planning, resource allocation etc.).

7. Conclusions

This paper presented an approach to facilitate management, sharing, and correlation of context between components in a large-scale pervasive computing environment. We have shown the significance of model-based interoperable systems in the attempt to avoid errors due to heterogeneity. Even when well-defined models are present, still the construction of real-world semantics is not guaranteed to be unique. We have demonstrated that in domains with sufficiently rich semantics, like ITS domain, it is useful to achieve model-based interoperability.

We observe that very little attention is given in literature to the impact of independent model construction on model-based interoperability. However, co-operative cars, ambient intelligence, the smart home, and the digital assistant are but a few applications under consideration requiring such autonomic, model-based interoperability. The evaluation of interoperability was made upon a set of evidenced characteristics. The paper directly discusses interoperability into the context-aware systems (CAS) domain, considering context variance and adjusting systems behavior due to the change. We have demonstrated that CASs have a common base architecture and that such architecture is evolving with the introduction of modern frameworks. We developed a context model (CM) based on object-oriented principles, which can be implemented in a dedicate context-aware intelligent transport (CAIT) framework.

In future research, we plan to study the impact of independent construction of models with real world semantics on machine interoperability. Specifically, we would like to study the influence of purpose, process, and context of model construction and variations on likelihood of conflicts due to heterogeneity and necessary boundary conditions on model-based interoperability interfaces and associated assessment techniques to ensure a system with robust interoperation capability in a heterogeneous and evolving environment.

R E F E R E N C E S

[1]. *Lee, D., Meier*, A Hybrid Approach to Context Modelling in Large-Scale Pervasive Computing Environments, Proceedings of the Fourth International ICST Conference on Communication System software and middleware, ACM Press, pp. 1 – 12, 2009

[2]. *Motta, R. C., de Oliveira, K.M., Travassos, G. H.*, Characterizing Interoperability in Context-aware Software Systems, VI Brazilian Symposium on Computing Systems Engineering, pp. 203-208, 2016

[3]. *Motta, R. C., de Oliveira, K.M., Travassos, G. H.*, Rethinking Interoperability in Contemporary Software Systems, 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems, pp. 9-15, 2017
[4]. *Zhong, Z., Gu, J.,Zhang, Y.Y.,Lin, X*, Enhancing applications interoperability in context management for practice tasks, IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support, pp.122-129, 2013
[5]. *Said, A. M., Soua, A., Abd-Elrahman, E.,Afifi, H.*, "Context-Aware Multi-modal Traffic Management in ITS: A Q-Learning based Algorithm", International Wireless Communications and Mobile Computing Conference, pp. 674-679, 2015
[6]. *Khan, Z., Kiani, S. L.,Soomro, K.*, A framework for cloud-based context-aware information services for citizens in smart cities, Journal of Cloud Computing: Advances, Systems and Applications, 3:14, pp. 1-17, 2014
[7]. *Feng, Y., An, X., Li, S.*, Application of context-aware in intelligent transportation CPS, 36th Chinese Control Conference, pp. 7577-7581, 2017
[8]. *Hendriks, T.,Wevers*, Abstraction Conflicts in Industrial Deployment of Model-Based Interoperability Standards, Proceedings of Conference on Systems Engineering Research, pp. 1-10, 2007
[9]. *Iordanopoulos, P.,Mitsakis, E., Rijavec, R., Kernstock, W.*, Requirements for Interoperable Intelligent Transport System Deployment in South East Europe, MPRA Paper No. 61568, Online at https://mpra.ub.uni-muenchen.de/61568/, 2015
[10]. *Zhang, S., Wu, Y., Wang, Y*, An Embedded Node Operating System for Real-Time Information Interaction in Vehicle-to-Vehicle Communication, IEEE 19th International Conference on Intelligent Transportation Systems, pp. 887-892, 2016
[11]. *Joshi, A., Gaonkar, P., Bapat, J.*, A Reliable and Secure Approach for Efficient Car-to-Car Communication in Intelligent Transportation Systems, International Conference on Wireless Communications, Signal Processing and Networking, pp.1617-1620, 2017
[12]. ISO/IEC/IEEE. Systems and Software Engineering Vocabulary. ISO/IEC/IEEE 24765, 2009