Cooperation Enabled Systems for Collaborative Networks

A. Luis Osório[1,3]; Luis M. Camarinha-Matos[2]; Hamideh Afsarmanesh[3]

[1]Instituto Superior de Engenharia de Lisboa (ISEL), Portugal, aosorio@isel.pt
[2]Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal (cam@uninova.pt)
[3]University of Amsterdam (UvA), Netherlands (h.afsarmanesh@uva.nl)

Abstract. A fast evolution to collaborative business models requires novel strategies for the development of collaboration-based information technology (IT) solutions. The complexity of building such solutions based on heterogeneous sub-systems requires a multidisciplinary approach, involving the perspectives of technology and business/management. In this direction, a novel approach is proposed, introducing the notion of Cooperation Enabled System as an autonomous and adaptive computational component as the base constructor to develop open (collaborative) IT solutions.

Keywords: Collaborative Networks Engineering; Cooperative Information Systems; Service Oriented Architectures.

1 Introduction

In a diversity of application domains we can notice that organizations are evolving towards becoming nodes of collaborative networks (CN) [1], [2] while participating in the offering of integrated business services. Ideally, these services are presented to clients in a way that makes transparent the participation of a diversity of organizations with their own information technology infrastructures.

The Brisa/Via-Verde tolling system and the national speed limit enforcement network are two examples of application domains that faced such challenge of adopting a collaborative networks strategy. The Brisa/Via-Verde case [4], [8] refers to the evolution of an existing tolling service to an integrated (multi-provider) business service, involving payments in parking lots and gas stations. One of the main problems faced in this case was the closeness of existing IT solutions; the electronic toll collection (ETC) subsystem was not prepared for the cooperation. An initial approach considered the development led by the ETC supplier as a unique possible integrator, but the adoption of proprietary processes and technology patterns resulted in an expensive "one-of-a-kind" solution. The national speed limit enforcement initiative, the SINCRO project from the Portuguese national road security authority had a quite different initial motivation. The objective of developing the network along different tendering phases suggested the adoption of a strategy aiming to guarantee the independence from the initial supplier(s). While the initial objective is to manage enforcement events directed to drivers, the adopted strategy is to establish a collaborative network involving public and private organizations.

In both Brisa/Via-Verde and SINCRO cases the aim is to offer ***integrated business services*** (IBS) implemented as collaborative processes which are supported by multiple stakeholders as nodes of a collaborative network (CN), Fig. 1. The adopted strategy in both cases features two key aspects:

i. The proposition of a *collaboration layer* (CL) [6] as a mechanism to isolate the processes and IT culture of each participating service provider;
ii. The proposition of a generic autonomous computational entity, the *collaboration enabled system* (CES) [6], aiming to handle the diversity at both organization's and collaborative network levels.

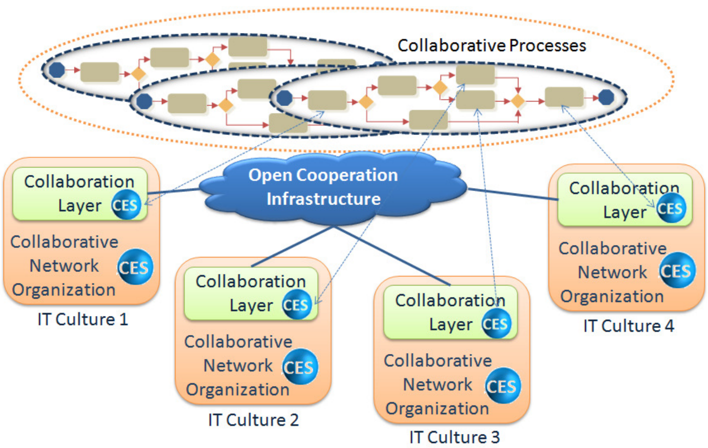

Fig. 1 – Collaborative processes invoking CES services in CN nodes

A main research hypothesis is to consider the adaptive computational component CES [6] as a key strategy to promote the openness of complex integrated systems. The elements of the suggested approach are detailed and discussed in the following sections. The IBS are designed under a collaborative business model perspective and supported by a set of collaborative business processes (CBP) [13] deployed and executed in the context of a collaborative network [7]. The collaboration enabled system (CES), is then introduced as the base executive component of an open framework and infrastructure for collaborative networks.

2 The Need for Collaboration-enabled Solutions

The challenge of developing open and integrated solutions for telematics domains such as the road tolling system, motivated the development of an open service bus, the ITSIBus [4], [9] as a first solution. In the beginning of the Brisa's project the road tolling systems were closed to competing suppliers besides the initial one. The independence from this unique supplier was then achieved through the development of an open service infrastructure involving a *service surrogate* as an adapter to make transparent the specificities of the different protocols from different road side equipments (RSE), as illustrated in Fig. 2.

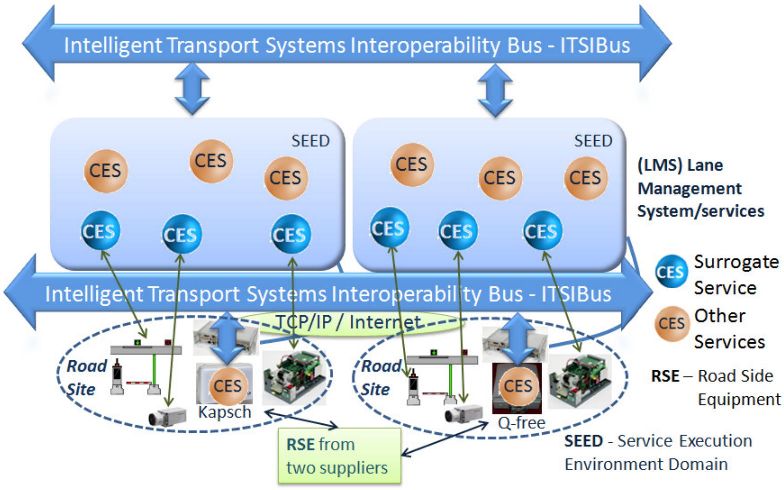

Fig. 2 – Adapter-based integration of systems from different suppliers

This approach followed a quite usual integration pattern, which avoids the need for changes in legacy systems. Nevertheless, this exogenous integration effort has the drawback of implying extra costs and operational risks considering that there is an additional computational system, in some cases involving also hardware. Moreover, the complexity of the aimed integrated collaborative solutions suggests novel approaches regarding the lower levels components (e.g. sensor, actuators to user interacting channels). While the concept of adapter or surrogate might coexist for the cases where the inclusion of legacy systems might be an advantage, a more promising approach is to develop a collaboration enabled systems framework as a unifying bridge between business requirements and technology. The idea is to introduce an abstract component language able to process requirements against standard reusable components. Such generic and adaptive computational component aims to contribute to an endogenous integration paradigm. By endogenous integration we understand the approach that promotes that each subsystem is, from its inception, developed for operation in a collaboration context. In Fig. 2 this idea is illustrated by the embedding a CES element into the Kapch and Q-free (the current Brisa's RSE suppliers) subsystems, making them collaboration-enabled, without the need for exogenous adapters. The strategy here was to promote the technological change without the need for extra costs from the Brisa's side.

The SINCRO project considers a central operations management unit (CGO) that will be responsible for the quality control of each speed limit enforcement event (a multimedia object with a picture, a timestamp and the measured speed). This CGO subsystem, managed by the national authority, will cooperate with other subsystems to process and validate enforcement events. The interaction with a system managed by the authority responsible for the vehicle registration is an example of such required collaboration among public organizations. Although a simplified scenario, this is an interesting case to illustrate the complexity of the integrated collaborative solutions. An initial question was how to develop the national network in different phases starting with the most dangerous road points and gradually adding new control zones. Without a definition of an open architecture the national authority could only state WHAT is required and it would be up to the tender winner to decide about the HOWs. The next question is how to promote the development of an open solution towards the reduction, if not the elimination, of market dependencies regarding the development, life cycle management and evolution of the planned complex collaborative solutions. Taking this scenario as example (similar to the tolling case) the main problems are:

- The existing speed enforcement systems are based on an integrated kinemometer (Doppler Radar or Laser), a camera and a controller typically implementing a proprietary protocol to access enforcement events (through TCP connection);
- A cabinet system (or rack) has a specific design and development usually provided by the supplier of the radar or by an integrator;
  An equivalent CGO system is offered by the market, usually by suppliers of radar systems as an integrated solution to manage the national speed enforcement networks.

In this case it might be incorrect to talk about integrated business services, since is more an enforcement on citizens. Nevertheless it is possible to consider the national authority as an integrated service provider in the sense that the citizen has gets a single contact to solve the enforcement events independently of the number of involved collaborating organizations. Please note that this case includes the participation of municipalities with their own speed enforcement infrastructures, police authorities as unique persons authorized to move radar systems among enforcement points, etc. Recently, these organizations have been trying to evolve to higher levels of collaboration but their heterogeneous IT solutions are a drawback to this process. A more ambitious vision, already under discussion, is to evolve to a pan-European collaborative network to manage traffic safety and other security issues, as part of a global strategy to a safer mobility in Europe. We might foresee a Portuguese citizen travelling through Europe to be contacted by a subscribed integrated service warning her/him about enforcement events generated along her/his is trip. We can also foresee some form of warning regarding the entrance into a driving environment with different rules.

3 The Concept of Collaboration Enabled System

The collaboration enabled system (CES) concept aims to contribute to a separation between the realization and the modeling of requirements [11]. The idea is thus to contribute to the consolidation of the model-driven development (MDD) approach [10], [12] as an agile and adaptive mapping between requirements and execution components. The purpose is to promote a system's, rather than software development thinking. This proposed systems' thinking requires an effort to generalize the implementation perspective making subsystems adaptable to a diversity of requirements and able to support the derivation of new subsystems with the required capabilities. The CES concept aims to support such flexibility, making it possible for a components repository to support process development tools during technology bindings when executive resources are associated to process activities.

A Collaboration Enabled System (CES) addresses three main challenges:

i. Support a model-driven and process-oriented development of open complex collaborative solutions;
   - Mapping of functional and non functional requirements to CES capabilities;
   - Model based selection and evaluation of CES's compositions;
   - A CES generalization or specialization should precede unmatched requirements resolution. This is more a methodological assertion, suggesting the developing of a new CES only when the previous step is not a possible option.

ii. Leverage the reutilization of IT systems by promoting adaptive cooperation mechanisms;
   - A CES system embeds a number of mechanisms making it ready for collaboration in different scenarios, namely, different technology bindings for equivalent capabilities, associated metadata able to be interpreted by potential clients both at development and running time stages.

iii. A contribution to make the management of complex collaborative solutions more agile and autonomous (dynamic adaptability to a network and systems management application/service).
   - Given the growing diversity of systems, there is a need for a unified systems' management considering both monitoring and maintenance under a unique service level agreement. While this is not the main focus of this research, the adopted architecture establishes the basis for an adaptive and intelligent integrated monitoring and maintenance processes automation.

Definition: A Cooperation Enabled System (CES) is an autonomous computational entity, made of interrelated components, defined as a tuple: CES = <SA, I, S, M, E, R, B>

where (Fig. 3):

- SA is a Self-Awareness subsystem embedding meta-data able to adapt the CES to a diversity of contexts; the SA subsystem manages CES capabilities;
- I is a set of Interfaces, considering an interface as a point of interaction or a cooperation point;
- S is the Security subsystem responsible for implementing the CES's security policies;

- M is the Monitoring subsystem aiming to implement the mechanisms making possible to plug the CES to an infrastructure management system (operational quality assurance);
- E is the Event management subsystem and it is responsible to support asynchronous interactions with CES clients;
- R represents the set of Resources necessary for the implementation behaviors. It might include configuration files, executable objects, user interface components among other resources to be used by the specialized behaviors;
- B embeds the business logic, implementing the modeled Behaviors that contribute to the declared capabilities.

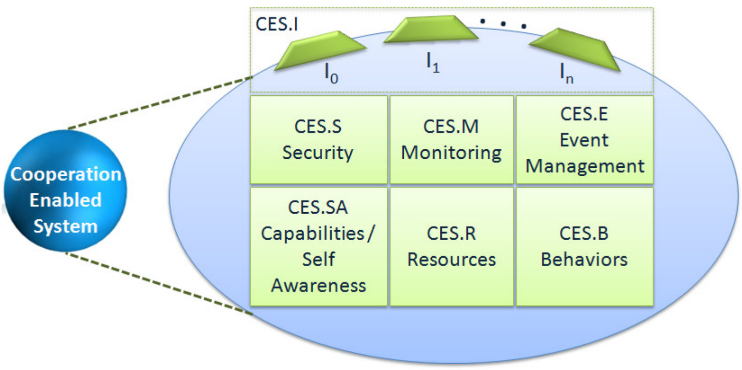

Fig. 3 – The Collaboration Enabled System (CES) structure

One important aspect of the proposed concept is the generalization of the implemented interfaces making it possible for a CES to adapt to a diversity of execution environments. For instance, the road side equipment (RSE) has autonomy to store vehicle identification events; when reestablishing the connection to the lane management service (LMS), there is a need for a potential exchange of a huge amount of stored data. One option is to provide a specialized file transfer protocol for a reliable transfer of such data. However, other functionalities can be available through web services of other inter-systems communication mechanism (FTP, RMI, .NET, etc.). The need to cope with scenarios involving multiple technologies also motivates the need to increase CES adaptability.

According to the above definition, all the CES components can be organized as a collection of generic modeling entities (GME), Fig. 4. In the proposed framework a GME object aims to represent any kind of information from data, knowledge or behavior. In this way, both the run-time system and any CES can dynamically adapt for the cooperation with another CES.

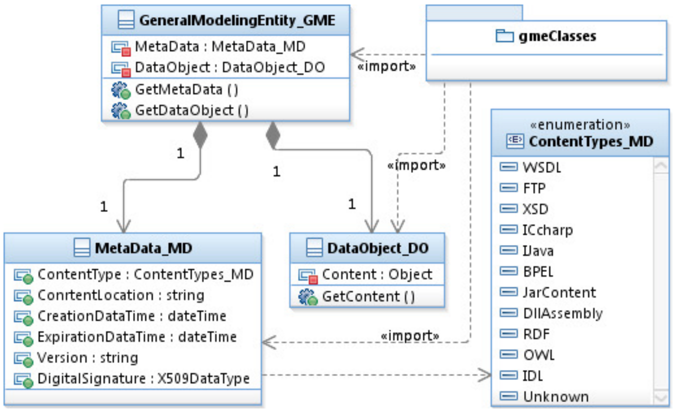

Fig. 4 – The GME UML model

A GME instance is composed by exactly of one MetaData_MD and one DataObject_DO objects. The attribute ContentType identifies a known structure of the content. When the value is Unknown, there is a possibility to infer its type by analyzing the content based on previous knowledge of existing specifications. This feature might be addressed considering a confidence degree to the inferred content type and according to the decision policies the result is adopted or discarded. The idea is to support semantic evaluation for the cases information is lacking to unambiguously interpret GME objects.

4 Revisiting the Tolling Application Example

A road side equipment (RSE) reads vehicle electronic identification from an on-board unit (OBU). When plugged, the CES/RSE looks for a CES services directory (which might be also a CES) and registers its I_0 interface. A lane controller, also implemented as a CES, discovers the RSE and, through its I_0 interface, gets the necessary context information to retrieve vehicle identification events to be used in cooperation with other lane subsystems to generate electronic toll transactions. At the lane level, a typical electronic toll collection system has a lane controller that coordinates events generated by vehicle detection and classification and an enforcement subsystem. The lane controller logic is modeled by a process definition, Fig. 5. Some initialization process loads the LMS process definition to a CES space capable of interpreting process definition. The idea is to assume a CES as embedding a process virtual machine able to interpret declarative business logic definitions (e.g. a BPEL interpreter or execution engine).

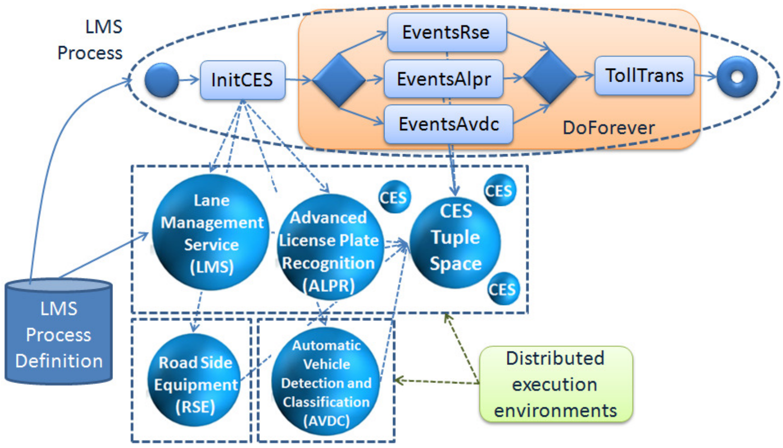

Fig. 5 – The lane management service (LMS) architecture

The proposed framework considers specialized system CES offering communication and coordination mechanisms for the entire solution. The cooperation enabling Tuple Space system (CES Tuple Space), Fig. 5, is responsible for the communication among CES. In the implemented system, this communication follows the Linda [3], [5] shared space paradigm. By using a tuple space service each CES has only a single interlocutor. A LMS subscribes events with a specific signature on the Tuple Space CES to access events from the RSE, AVDC or ALPR subsystems. A strategy like this simplifies the implementation of a CES considering it does not need to be aware of the location of the other cooperating CES.

The CES proposal is grounded on the results from the adoption by Brisa of the ITSIBus service oriented (SOA) infrastructure [9]. The adoption of some of the features proposed in CES like the service monitoring, has contribute to a reduction of the total cost of ownership (TCO) of the deployed technology. It has also contributed to make simple and more competitive de evolution for new services as the case of the recent deployment of a new self-service tolling lane type (eTOLL).

5 Revisiting the Speed Limit Enforcement Example

The SINCRO project has established an initial functional granularity based on three autonomous subsystems being modeled as CES services:

i. The central operations management (CGO), which is modeled according to a business process modeling basis. The executive CES interprets the process representing the CGO responsibilities;
ii. The cabinet subsystem embeds a CES responsible for the management of the rack even when no radar subsystem is plugged in;
iii. The radar subsystem is modeled as a CES and is responsible to deliver speed enforcement events to the CGO.

In a first approach, single network management protocol (SNMP), file transfer protocol (FTP) and web services technology have been adopted to promote the open interfaces for the planned subsystems. A management information base (MIB) for the cabinet and radar subsystems were already defined to support the monitoring perspective of each the subsystems. A reference implementation to validate the proposed interface and models is under development. The strategy is to maintain a pragmatic development, close to the potential suppliers' innovation capabilities and, at the same time, develop a theoretical thread able to rupture processes and technology patterns towards the adoption of the proposed adaptive CES.
An eclipse plug-in is planned as a model drivel development (MDD) tool to manage the CES bindings to the business process activities.

6 Conclusions

The development of complex integrated collaborative IT solutions is an open and challenging task. Existing complex IT solutions are typically closed as they depend on a single responsibility (single provider). The ongoing development of a new generation of open electronic toll collection solution under a research project sponsored by Brisa company illustrates the existing problems. This research was motivated by the extension of the offered service to the payment in parking lots and gas stations and by the difficulties for the existence of a unique technology supplier (closed solution). The project is now evolving to the evaluation of the cooperation enabled system (CES) concept where an adaptive systems framework embedding service capabilities is being proposed.

The main objective of the proposed CES framework is to support the technology bindings for process definitions independently of considering intra-organizational processes or collaborative processes (or cross-organizational processes. The purpose is to offer a model-driven approach to the development of IT solutions (process oriented developments). This means that suppliers are invited to develop their systems able to plug into compositions of systems leading, in this way, to open integrated solutions. The openness of such complex IT solutions is not an easy issue. The responsibility constraint related to the possibility of any subsystem being replaced by a competing one, requires further research. For less critical systems, the interoperability certification is a mechanism being developed to guarantee the replaceability of a subsystem by an equivalent one (from a competing supplier).

The proposed CES structure is also being validated in a governmental authority application for the development of a national speed limit enforcement network, the SINCRO project, promoted by the Portuguese National Authority for Road Security (ANSR).

Acknowledgements. This work was partially supported by BRISA Innovation and Technology, and Galpgeste companies and National Authority for Road Security (ANSR) through a research and development grant.

References

1. Afsarmanesh, H., Msanjila, S.: EPAL vision 2020 for active ageing of senior professionals. In Luis Camarinha-Matos, Xavier Boucher, and Hamideh Afsarmanesh, editors, *Collaborative Networks for a Sustainable World*, volume 336 of *IFIP*, pages 60–72. Springer Boston, 10.1007/978-3-642-15961-9_6, (2010).
2. Camarinha-Matos, L. Afsarmanesh, H.: Active ageing roadmap: a collaborative networks contribution to demographic sustainability. In Luis Camarinha-Matos, Xavier Boucher, and Hamideh Afsarmanesh, editors, Collaborative Networks for a Sustainable World, volume 336 of IFIP, pages 46–59. Springer Boston, 10.1007/978-3-642-15961-9_5 (2010).
3. Ciancarini P.: Coordination languages for open system design. In *Computer Languages, 1990., International Conference on*, pages 252 –260, March (1990).
4. Gomes, J. S., Jacquet, G., Machado, M., Osório, A. L., Gonçalves, C., Barata, M.: An open integration bus for efc: The its-ibus. In *ASECAP conference*. ASECAP-2003, 18 - 21 May 2003 in Portoroz, Slovenia, May (2003).
5. Holvoet, T., KielmannT.: Behaviour specification of active objects in open generative communication environments. In *System Sciences, 1997, Proceedings of the Thirtieth Hawaii International Conference on*, volume 1, pages 349 –358 vol.1, January (1997).
6. Osório A. L., Afsarmanesh, H., Camarinha-Matos, L.: Open services ecosystem supporting collaborative networks. In Ángel Ortiz, Rubén Franco, and Pedro Gasquet, editors, *Balanced Automation Systems for Future Manufacturing Networks*, volume 322 of *IFIP Advances in Information and Communication Technology*, pages 80–91. Springer Boston, 10.1007/978-3-642-14341-0_10 (2010).
7. Osório, A. L., Camarinha-Matos, L.: Towards a distributed process execution platform for collaborative networks. In Weiming Shen and Luis Camarinha-Matos, editors, *Information Technology For Balanced Manufacturing Systems*, volume 220 of *IFIP International Federation for Information Processing*, pages 233–240. Springer Boston, 10.1007/978-0-387-36594-7_25, (2006).
8. Osório A. L., Camarinha-Matos, L., Gomes, J. S.: A collaborative network case study: The extended "viaverde" toll payment system. In Luis Camarinha-Matos, Hamideh Afsarmanesh, and Angel Ortiz, editors, *Collaborative Networks and Their Breeding Environments*, volume 186 of *IFIP International Federation for Information Processing*, pages 559–568. Springer Boston, 10.1007/0-387-29360-4_59 (2005).
9. Osório, A. L., Gonçalves, C., Araújo, P., Barata, M., Gomes, J. S., Jacquet, G., Dias R.: Open multi-technology service oriented architecture for its business models: The itsibus etoll services. In *Collaborative Networks and Their Breeding Environments*, volume 186 of *IFIP*, pages 439–446. Springer Boston, 10.1007/0-387-29360-4_46 (2005).
10. Pastor, O., España, S., Panach, J. I., Aquino, N.: Model-driven development. *Informatik-Spektrum*, 31:394–407, 10.1007/s00287-008-0275-8 (2008).
11. Scheer, A.W., Hoffmann, M.: From business process model to application system" developing an information system with the house of business engineering (hobe). In *Advanced Information Systems Engineering*, volume 1626 of *Lecture Notes in Computer Science*, pages 2–9. Springer Berlin / Heidelberg, 10.1007/3-540-48738-7_2 (1999).
12. Winkler, S., Pilgrim, J.: A survey of traceability in requirements engineering and model-driven development. *Software and Systems Modeling*, 9:529–565, 10.1007/s10270-009-0145-0 (2010).
13. Zhao, X. Liu, C.: Tracking over collaborative business processes. In Schahram Dustdar, José Fiadeiro, and Amit Sheth, editors, *Business Process Management*, volume 4102 of *Lecture Notes in Computer Science*, pages 33–48. Springer Berlin / Heidelberg, 10.1007/11841760_4 (2006).