46

# OPEN MULTI-TECHNOLOGY SERVICE ORIENTED ARCHITECTUR FOR "ITS" BUSINESS MODELS: THE ITSIBus ETOLL SERVICES

A. Luís Osório[1], Carlos Gonçalves[1], Paulo Araújo[1], Manuel Barata[1]
J. Sales Gomes[2], Gastão Jacquet[2], Rui M Dias[2]
[1]ISEL, Instituto Superior de Engenharia de Lisboa, GIATSI research group, Portugal
*{aosorio, cgoncalves, paraujo, mbarata}@deetc.isel.ipl.pt*
[2]*BRISA, Auto-estradas de Portugal, DIT Innovation and Technology Department, Portugal*
*{Jorge.Gomes, Gastao.Jacquet, Rui.Dias}@brisa.pt*

*The development of integrated solutions made of systems based on different technologies, adopting different implementation approaches and different versions is a complex challenge. The lack of standards or differences in implementations when they exist, are important obstacles to the construction of integrated, flexible and agile solutions. The incorporation of systems from different vendors and systems that evolve to answer innovation processes, suggests the advantage for a multi-technology systems strategy. This paper discusses the ITSIBus approach based on, a multi-technology service oriented infrastructure where specialized pluggable systems run services following a peer to peer architecture. The multi-technology approach is based on the discussed System Broker concept. The agility required by a crescent number of collaborative business process requires an advanced flexibility from the ICT technological infrastructure. The ITSIBus approach is also discussed as a grounding platform to support enterprise collaborative networks considering that services in different companies are based on different technologies.*

## 1. INTRODUCTION

The integration of functions executed across networks of enterprises requires an advanced integration level for the internal enterprise processes. The emergent concept of enterprise collaborative networks (Camarinha-Matos, 2003) needs a long way to be adopted by existing de facto networked enterprise. These networks are deeply grounded on file transfer and are far to answer collaborative process requirements. There is an effective need for a tighter integration strategy among network members. This movement from a loose connection approach based on file transfer or even physical document transfer, to a transactional collaboration based on a tight integration approach, requires a new strategy from both technological and business process perspectives. However, before the discussion focused on enterprise network collaborative level, there is a need to establish at enterprise internal model (Camarinha-Matos, 1999) an infrastructure agile enough to cope with systems integration requirements. The integration among disparate systems developed on different technologies and supplied by different vendors is a main challenge for the agile collaborative enterprise network. This paper discusses the ITSIBus approach (Gomes et. al. 2003-b) (Osorio, et. al. 2004) considering its implementation on the

JINI (Flenner, 2002) platform with enhanced features aiming to interoperate with systems developed on other platforms like .NET (Rammer, 2005).

The ITSIBus was motivated by the extended ViaVerde toll payment business model (Osorio, et. al. 2003). The evaluated requirements have pointed out for the need to integrate a number of systems in different technological stages and versions – a strategy grounded on multi-vendor systems. The ITSIBus architecture proposes a set of standard services for specific functionalities like toll collection based on DSRC/RFID technology, vision based licence plate recognition and automatic toll collection based on card payment - ETOLL, among other systems. Beyond the specialized functionalities (generic services) the ITSIBus establishes a set of core services defined in WSDL and able to be bound to different platforms. The interoperability among services running on different technologies (platforms) is guaranteed by a special type of system, the SystemBroker. The SystemBroker provides inter-technology service interoperability. It offers a transparent access to services implemented in one technology, for instance on JINI, from services implemented in another technology, for instance on .NET.

The strategy aims to establish a technological infrastructure able to integrate services from concurrent providers (multi-vendor). Furthermore it considers that the services are plugged on a plug-and-play basis (Allard, 2003) and follow ITSIBus management specification. The adopted strategy is based on the assumption that holistic systems are made of heterogeneous components or systems independently of the underlying technologies. The strategy is quite different from the CORBA initiative where an object request broker and a binding mechanism for different languages were created for objects (or execution components) developed in different languages. The common denominator was the service definition language IDL (Interface Definition Language) with a set of standard bindings to different programming languages. The strategy proposed by ITSIBus considers a uniform service definition using the WSDL language as a common service definition language. Even if some extensions are needed to define concepts like event registration and subscription, the strategy is grounded on other efforts to use WSDL with or without extensions to be used as a reference service modelling language.

The paper discusses the System Broker strategy as a way to preserve development culture and at the same time offer a wide range of system developers the opportunity to integrate their solutions towards a holistic and open technological platform. The ETOLL System is discussed as a case study of such approach. The ETOLL system is an automatic tollbooth where car drivers can pay toll using payment cards, coins and other payment forms all done without the presence of an operator. This system was developed in C++ and based on Windows operating system. The challenge was to offer ETOLL development team an easy access to ITSIBus services and system concepts without too much interference on ongoing developments. For that a System Broker is being developed to accept ITSIBus service discovery, service registration, service calls, and call-back management mechanisms in order to interoperate with services implemented in other platforms than JINI (Osorio, et. al. 2004).

The paper discusses also a model based development approach to ITSIBus systems and services based on Eclipse platform (www.eclipse.org). A plug-in to Eclipse was developed to speed-up solutions development based on a set of wizards guiding code generation and providing a development framework where developers are focused on business logic and not on intrinsic of the underlying technology concepts.

Since the beginning of the ITSIBus project a special concern was dedicated to the "leaning curve" required to understand advanced systems and concepts like those proposed by ITSIBus and those associated to the underlying technologies. Even if JINI is not a complex platform it embeds a large number of new concepts and it is a huge framework requiring specialized knowledge do deal with. To facilitate service development, a plug-in for the Eclipse platform is being developed to guide toll management system developers to create systems, services and client applications. The client application aims to test ITSIBus systems and services during developments. The main goal for such a kind of workshop is to speed-up the development of new systems and services by focussing developers on service business logic. The objective is to offer experts on toll management an advanced tool where services and service integration on complex systems can be developed from models, reducing to a minimal the code needed to be written.

With ITSIBus development workshop a developer does not need to be an expert on JINI or in any other underlying technology selected to support ITSIBus concepts.

The ITSIBus is discussed as a grounding ICT infrastructure to support higher integration levels, namely that concerned with collaborative enterprise business processes. In fact the motorway management toll involves services to car drivers beyond the motorway management company (Osorio, et. al. 2004). In the case of the BRISA Company, the toll was extended for payments in car parking areas and gas stations. The ITSIBus infrastructure is also discussed as a grounding base for the supporting of some advanced challenges namely those related with the enterprise collaborative networks. It is a requirement of the toll technological infrastructure that collaborative companies access toll systems to audit toll collection processes. This requires an innovative approach to manage network level business processes. The ITSIBus is being considered as an underlying infrastructure to manage such cross company processes on a tight integration approach perspective.

## 2. THE MULTI-TECHNOLOGY APPROACH FOR ITSIBUS

The rapid evolution on ICT occurred during the last decades has put many challenges to ICT developing companies. Development teams have to deal with different equivalent technologies and with different frameworks and tools for each technology. There is a trend for some unification of key computational concepts but when implemented in different technologies it might originate interoperability problems. The utilization of Web Services is a way to get a common path but it is not a solution for all the challenges.

For ICT companies the challenge is how to deal with different equivalent technologies each one with its intrinsic and its corresponding learning curve. In fact, this is one of the bottlenecks when considering evolution of enterprise systems. When adopting new technologies companies need to make investments on training. Furthermore they have to support a lost of productivity until the development team gets ready with the new concepts and tools.

The ITSIBus strategy to cope with this multi-technology challenge is to establish a SystemBroker concept aiming to play a flexible bridge role connecting services developed in different technologies. In the area of network management systems

Grabowski (Grabowski, 1999) has also proposed an integration architecture of multi-technology management systems. The ITSIBus is a service-oriented architecture based on a community of peer systems. A system implements a set of basic services named core services (security, management and plug-and-play). The services are defined in WSDL and are a contribution from Brisa to standardize similar services exported by roadside systems (equipments). In the context of ITSIBus the services under standardization are the DSRC/RFID car identification service, car detection and classification (AVDC) service, car plate automatic recognition (ALPR) service, tollbooth system services and ETOLL semi-automatic toll system services. All these systems export services and generate events establishing this way a flexible infrastructure to interconnect peer services (through systems) providing synchronous or asynchronous association (Figure 1).

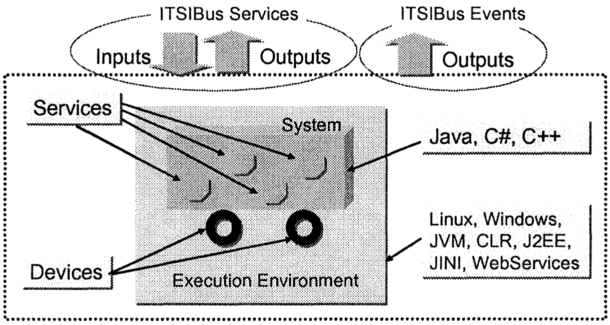


Figure 1 – Basic constructs of the ITSIBus infrastructure

Even if different underlying technologies where evaluated like Web Services and JXTA (Brendon, 2002), the JINI platform was the selected for the first reference implementation of ITSIBus. In fact, the productization process coordinated by Brisa and with the collaboration of WhatEverSoft, a Portuguese software development company, adopted JINI as the underlying technology to develop ITSIBus services. Nevertheless, some of the involved systems were integrated through adapters, i.e., service implementation is in the reality an adapter to establish the connection between ITSIBus infrastructure and the legated roadside system. This happen with the DSRC system already developed by Q-free and Kapsch companies as ITS (Intelligent Transport Systems) technology suppliers. Even if this was a successful strategy, the need for an adapter is a source for entropy mainly as far as evolution is concerned. In fact, the need for adapters is a common strategy to integrate disparate systems but it is also a source of problems. As already discussed, the experience with the added complexity reflected in a low quality of services and high costs of maintenance and evolution justifies the need for a different approach.

Considering that it is not advisable to impose a change in suppliers' technological culture the adopted strategy was to follow a multi-technology approach. The objective is to push for a common agreement at service definition level using standards like WSDL, UML and other more semantic oriented interoperability facilitators, considering the technological heterogeneity as a challenge for the ITSIBus platform. To promote technological interoperability based on multiple technologies the System Broker concept was proposed (Figure 2). The System Broker is a special ITSIBus System dedicated to offer platform level

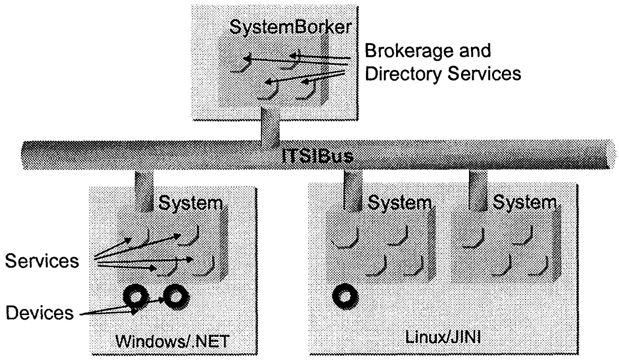


Figure 2 – Multi-technological peer-to-peer service oriented infrastructure

services like the Reggie Service required by JINI to implement system and service discovery and advertise mechanisms. It is assumed that systems developed in different technologies are grounded on the basic TCP/IP network protocols; this is the exclusive mandatory requirement for information exchange between heterogeneous systems. The System Broker is a multi facet system embedding components that know how to talk to services in all the accepted underlying technologies. Internally these components cooperate to dynamically exchange service calls and event routing among the heterogeneous systems/services. A deployed ITSIBus infrastructure must include at least one System Broker, however it can have more than one running, what might contribute to increase availability because the System Broker is a critical point of failure.

When a system is powered-on it registers itself on a System Broker and implicitly all its services are also registered. Depending on the underlying technology of each particular system it is necessary to guarantee the availability of discovery and advertising mechanisms enabling each system and service to known the other ones. As an example, if the underlying technology is JXTA all the Systems/services available in other underlying technologies must be represented by JXTA services embedded on a System Broker. When a JINI service looks for other services, it sees all of them independently of the technology in which they are implemented. When a JINI service accesses a service developed in JXTA the access goes through the System Broker and is the proxy service that bridges the call and replies to the JINI client service.

A first prototype involving services developed exclusively in JINI and implementing a System Broker is already implemented. To demonstrate the System Broker strategy a first prototype integrating ITSIBus services on JINI and services on .NET is being implemented. The selected scenario for this challenge is the ETOLL lane payment system based on a semi-automatic toll payment mechanism. In the next chapter, some details about ETOLL developments are presented.

# 3. THE ETOLL SYSTEM AND SERVICES EXAMPLE

The ETOLL system is another example of roadside equipment playing a tollbooth role without a person operating the tool payment. An existing system developed by Brisa initially implemented in C++ was the starting point. Considering the interest of its developing company, the evolution of this system was considered to plug to the ITSIBus resorting to the .NET technology rather then JINI. The decision taken was based on the existing ground culture within the company.

The ETOLL system incorporates a LMS system with a lane coordination service. This service manages other systems like DSRC, AVDC, LIT and a gate, responsible to interact with toll users (Figure 3). When using this kind of lane the driver selects the payment using a payment card through the LIT system or else using money (coins or paper) and in this case, the money input/change system is used. This type of lane includes also a gate to be opened when the payment succeeds, also implemented through a specialized service.

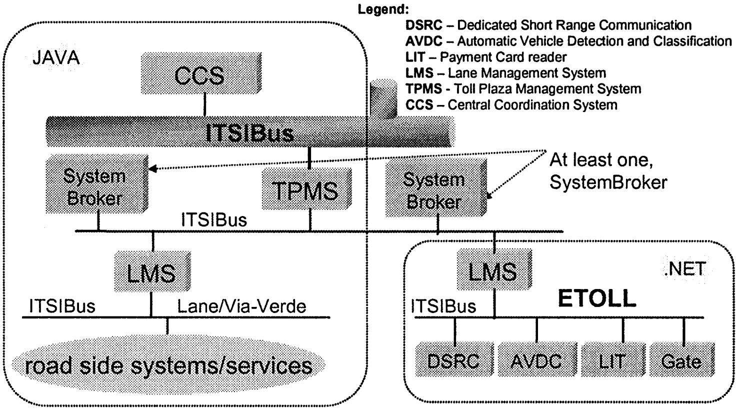


Figure 3 – Architecture of the toll technological infrastructure

What is important to the proposed discussion in this paper is not if the ETOLL subsystem follows the ITSIBus architecture but rather its integration to the toll management infrastructure. In fact, the prototype under development is considering the implementation of the ETOLL system following the ITSIBus architecture bound to the .NET technology suit. However, what is important is to promote interoperability according the adopted system/service granularity. In the case of the ETOLL system, the plugability is discussed at LMS level, i.e., the ETOLL system is a closed system (on .NET) that will integrate to TPMS and the other LMS systems developed on JINI.

All the systems/services are registered on a System Broker. The directory service needed to embed into System Broker depends on the adopted underlying technology. If the JINI platform is the adopted technology, the *Reggie* service needs to be running for services to be registered and discovered. The System Broker creates proxy services for each relation between services on different technologies. By now, only two underlying technologies were evaluated. When more than one underlying technology is considered the proxy needs to know how to bridge among different technologies. If JINI, .NET and JXTA are adopted a proxy service on a System Broker must route calls from any technology to implementations in any other technology (Xu, 2004). This is an open question under evaluation considering not only latency introduced by proxies but also dynamic adaptability to service definition changes. In order to facilitate developments on ITSIBus framework a set of tools based on Eclipse platform (Gallardo, 2003) are being developed. In (Figure 4) a partial view of the developed wizards to conduct developer on ITSIBus code generation is shown.

The first version of an ITSIBus tool suite (ITSIBus workshop) considers a guided (automatic) generation of Systems, Services and a Client application. The concept of client application was introduced to make developments easier by facilitate tests and debugging.

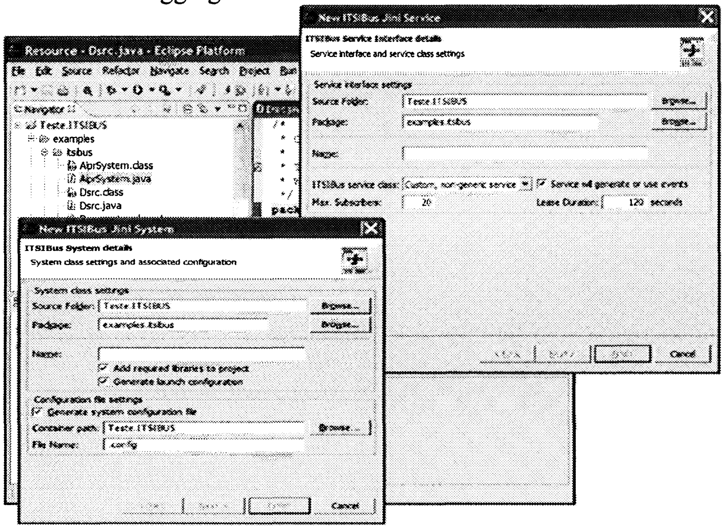


Figure 4 – Partial view of the ITSIBus Eclipse based development framework

The main objective of our research strategy is to contribute to reduce complexity when integrating systems based on different underlying technologies. Even if some technologies are being considered as a way to plug disparate systems like what is happening with Web Services, there is a generalized consensus that evolution will not happen under technology uniformity. Some specific requirements like those related with real-time systems, the reutilization of acquired knowledge and experience, cultural factors and other aspects, are some of the motivation contributing to accept the diversity of technologies beyond methodologies and tools. In fact the diversity contributes to an increased competition and to speed-up innovation processes. Therefore, systems evolution associated to an increased holistic approach are better supported by a multi-technology approach.

Considering enterprise collaborative challenges, it is mandatory to create such integrated systems at enterprise level in order to be prepared to collaborate on a service based considering process oriented transactions and not the already most used, file based collaboration.

## 4. CONCLUSIONS

The collaborative enterprise in the ICT era needs to achieve an added agility to cope with the dynamics of the collaborative processes. The ITSIBus technological infrastructure and the associated framework is a contribution towards a toll management system based on peer services. The services run on systems (as execution containers) connected on a plug-and-play basis. Furthermore, systems can be implemented on different technologies promoting this way an evolutionary strategy founded not only on existing knowledge but also on technology evolution (and competition). It is a reality that the standards and open reference implementations are facilitating the development of complex system by making easier the integration of heterogeneous components. Nevertheless, the adopted multi-technology strategy makes possible the inclusion of systems developed in different technologies increasing the systems integration potential. Furthermore, the ITSIBus specification is an open specification and there is a plan to transform the reference implementation in an open source initiative aiming to promote ITSIBus adoption by the ITS ICT industry.

The development of a framework where repetitive and complex tasks are automated through wizards helps the adoption of ITSIBus by ITS systems suppliers. The adoption of Eclipse follows a crescent consensus around the need for a common development platform extensible and used along the technological systems life cycle. The available tools present uniform concepts along the conception, design, development, deployment and operation cycles what might help to get a holistic perspective of enterprise ICT systems. This enterprise consistency is a key step for companies to get involved in collaborative networks on a wide service based collaborative infrastructure.

**Acknowledgements**. This work was partially supported by BRISA group, through the research and development ITSIBus project. The research groups GIATSI from ISEL-DEETC in collaboration with DIT, the Innovation and Technology Department of BRISA are developing the work. We also acknowledge the valuable contributions from Filipe Braz, Bruno Basílio, Joaquim Pereira, Rui Gonçalves and Rui Lopes. We also thank the valuable contributions from the WhatEverSoft team coordinated by José Rui Soares.

# 5. REFERENCES

Allard, J.; Chinta, V.; Gundala, S.; Richard, G.G., III - Jini meets UPnP: an architecture for Jini/UPnP interoperability, Applications and the Internet, 2003. Proceedings. Symposium on , 27-31 Jan.03.

Brendon, J. Wilson – JXTA, New Riders Publishing; ISBN: 0735712344, 1st edition, June 15, 2002.

Camarinha-Matos, L. M. - Infrastructures for virtual organizations – where we are, in Proceedings of ETFA'03 – 9th Int. Conf. On Emerging Technologies and Factory Automation, ISBN 0-7803-7937-3, pp 405-414, Lisboa, Portugal, 16-19 Sept 2003.

Camarinha-Matos, L.M.; Afsarmanesh, H. - The PRODNET architecture, , Proceedings of PRO-VE'99 - IFIP Int. Conf. on Infrastructures for Virtual Enterprises (L.M. Camarinha-Matos, H. Afsarmanesh–Eds.), Kluwer Academic Publishers, ISBN 0-7923-8639-6, pp. 109-126, Oct 99, Porto, Portugal.

Flenner, Robert - Jini and Javaspaces Application Development, Sams book Edition, 2002.

Gallardo, David; Burnette, Ed - Eclipse in Action: A Guide for the Java Developer, by, Robert McGovern, Manning Publications, ISBN: 1930110960, May 15, 2003.

Grabowski, C.; Bjerring, L.; Peykarimah, S.; Sorensen, L.B.; Bracht, R.; O'Hara, J.; Wade, V.P.; Karp, H. - Integration architecture of multi-technology management systems, Integrated Network Management, 1999. Distributed Management for the Networked Millennium. Proceedings of the Sixth IFIP/IEEE International Symposium on , 24-28 May 1999.

Osório A. L., Abrantes A. J., Gonçalves J. C., Araújo A.; Miguel J. M., Jacquet, G. C.; Gomes, J. S. - Flexible and Plugged Peer Systems Integration to ITS-IBUS: the case of EFC and LPR Systems, PROVE'03 – 4th IFIP Working Conference on Virtual Enterprises, published by Kluwer Academic Publishers, ISBN: 1-4020-7638-X, pages 221-230, 2003-b.

Osório, A. Luís; Barata, M. Martins; Abrantes, A. Joaquim; Gomes, J. Sales; Jacquet, G. Costa - Underlying ITS Business Processes with Flexible and Plugged Peer Systems: the Open ITS IBus approach, PROVE'03 – 4th IFIP Working Conference on Virtual Enterprises, published by Kluwer Academic Publishers, ISBN: 1-4020-7638-X, pages 221-230, 2003-a.

Osório, A. Luís; Osório; L., Barata, M.; Gonçalves, C.; Araújo, P.; Abrantes, A.; Jorge, P.; Gomes, J. Sales; Jacquet G.; Amador, A. - Interoperability among ITS Systems with ITS-IBus framework, BASYS 2004, Vienna, Austria 27-29 September 2004.

Rammer, Ingo; Szpuszta, Mario - Advanced .NET Remoting– Apress Editions, ISBN: 1590594177, 2005.

Xu, Jiangyun; Du, Weichang - Software brokers for quality of services in service-oriented distributed systems, Communication Networks and Services Research, 2004. Proceedings Second Annual Conference on, 19-21 May 2004.