The e-HUB Evolution: From a Custom Software Architecture to a Software-as-a-Service Implementation

David Concha, Javier Espadas, David Romero, Arturo Molina*

Tecnológico de Monterrey, Campus Monterrey & Ciudad de México, México
a00262912@itesm.mx, mijail.espadas@itesm.mx, david.romero.diaz@gmail.com, armolina@itesm.mx

Abstract: New technological advances have enabled the creation of distributed collaborative industrial networks giving origin to new collaborative e-business models such as the Virtual Organisation (VO). These industrial networks require new Information and Communication Technologies Infrastructures (ICT-I) playing the intermediary role as the enablers of interoperability among their participant organisations. Furthermore, these networks require coordination and cooperation mechanism among its members which in turn can be supported through specific vertical applications deployed over a common platform. PyME CREATIVA was a project defined with the aim of playing the role of a service provider of an open technological platform named the e-HUB, providing a set of collaborative electronic solutions named the e-services, to support collaborative business processes among Small and Medium-sized Enterprises (SMEs). Till today, the e-HUB platform has proved to be a complete IT architecture able to satisfy SMEs technological requirements to get involved in collaborative business opportunities, nevertheless with the technology progress new opportunities appear to improve the way these e-services are delivered to the SMEs. This paper provides an insight into the benefits and implications of evolving the e-HUB architecture into new one based on Software-as-a-Service (SaaS) premises.

Keywords: e-HUB, e-Services, Enterprise Integration, Software Architecture, Software-as-a-Service.

1. INTRODUCTION

In the 21st Century, new technological advances have enabled the creation of distributed collaborative industrial networks, supporting the emergence of novel organisational forms such as the extended enterprise and the virtual organisation. These new collaborative network structures aim to capitalize on the individual capabilities of their members, mainly Small and Medium-sized Enterprises (SMEs), by bringing them together in a temporary arrangement (e.g. coalition) in order to share skills or core-competencies and resources to better respond to (collaborative) business opportunities (Camarinha-Matos & Afsarmanesh, 2004). In this scenario, Information and Communication Technologies (ICTs) are seen as a key enabler for collaboration among SMEs. Moreover, these new collaborative e-business models require new coordination and cooperation mechanisms supported by computer networks. Therefore, new ICT-infrastructures (ICT-I) are need for the effective implementation of collaborative industrial networks, playing the role of ‘inter-operators’ and ‘service providers’ to guarantee the interoperability between distributed and heterogeneous applications and offer complementary e-services needed for supporting collaborative business processes among SMEs (Camarinha-Matos & Afsarmanesh; 2004; Rabelo & Gusmeroli, 2008).

As a respond to the demand for new ICT-infrastructures, the PyME CREATIVA project was born with a grant of the Interamerican Development Bank with the purpose of developing a novel ICT-I that was known later as the e-HUB (Integrated e-Services Center for Virtual Business) with aim of supporting e-partnerships (e.g. virtual organisations creation) and e-commerce between SMEs stakeholders (e.g. customers and suppliers).

The technological requirements for the e-HUB implementation were to develop an open, low cost and easy-to-access/use technological platform. Therefore, PyME CREATIVA project was focused on producing a low cost ICT-I taking advantage of open source technologies available. Additionally, the e-HUB was created thinking on an e-business environment that could foster the creation of collaborative industrial networks among SMEs by means of a common technological platform. The e-HUB was also provided with a group of collaborative electronic solutions known as ‘e-services’, which intend to reduce the critical troublesome among SMEs collaborative traditional limits, enabling the extension of their business processes into collaborative ones (Molina et al, 2006). The e-HUB vision was then to allow SMEs to execute trading processes, purchase orders, supply chain management, request for quotations, among other e-business transactions with others SMEs through the e-HUB (Jimenez & Espadas, 2006). The following e-services were developed and integrated within the e-HUB architecture platform (Molina et al, 2006):

- Brokerage. This e-service provides support for e-business and the exploitation of collaborative business opportunities through the creation of temporary alliances of enterprises named: Virtual Organisations (VOS). It supports the mediation processes between customers and suppliers. The e-brokerage service helps with the matchmaking between customers’ needs against a set of suppliers’ offerings. It also provides a communication channel to perform a transparent and audible e-negotiation process. The following features are supported by the e-brokerage service: (1) customer/supplier information management - consisting in a fully structured e-catalogue that dynamically aggregates information and presents it in an appropriate view accordingly to the user needs; and
(2) e-negotiation - supporting the negotiation activities which are sometimes considered as the most critical part of concreting a business opportunity.

**e-Supply.** This e-service implements a set of technologies for integrating manufacturing execution processes (among SMEs), order processing tracking capabilities and customer/supplier relationship management functionalities. Additionally, it provides logistics optimization and supply chain integration systems for managing a Web-based manufacturing execution system.

**e-Marketing.** This e-service enables an easy deployment of an intelligent and customizable enterprise portal with its customer relationship management system. The e-marketing service responds to the SMEs need for having an adequate technology to support their presence over the Internet. Basically, the e-marketing service offers an interface for an easy portal configuration and content management to provide each SME with its own Web portal, customized product/service e-catalogue and customers and suppliers information database.

**e-Productivity.** This e-service incorporates technologies for diagnosing, planning and monitoring the SMEs performance according to productivity and benchmarking indicators. The use of benchmarking as a core tool for the e-productivity service allows an enterprise to monitor the competitive environment for best practices. Therefore, SMEs can use this tool to measure their productivity level and track their improvements against local and world wide best practices.

**e-Engineering.** This e-service provides a collaboration engineering environment through an integrated product design and development framework. The e-engineering service intends to offer a step-by-step methodology as a guiding tool for a product development process. The processes and activities (workflow) are defined based-on an integrated product, process and manufacturing system development framework (see Aca et al, 2004), introducing the best practices in a product development process. The e-engineering service integrates all the relevant workflows normally carried out in product development process in order to offer a structured way of designing, innovating and developing (new) products. Those SMEs with a lack of properly implemented product development processes and qualified engineers in design and manufacturing are strongly benefited from this e-service.

The e-HUB concept covers three ICT-I innovation paradigms according to Molina et al (2006): (1) the e-HUB platform as a service provider serves as the gate to a wide range of value added e-services (applications) for SMEs; (2) the e-services implementation under a service-subscription delivery model demonstrates its impact and benefits to the SMEs by providing them with relevant technology for improving their competitiveness without worrying about the technological requirements for the software deployment; and (3) the creation of new e-business models for SMEs based-on collaborative industrial networks, supported by computer networks.

The e-HUB in its first implementation brought the e-services to SMEs under the service-subscription delivery model, but at that time the technologies available for such model were in their very early maturity stage. There were no pure service-subscription platforms to enable an easy creation of e-services meant for serving many tenants (e.g. SMEs). In contrast, today is relatively easy to find such platforms offering built-in multi-tenancy, usage metering, subscription models, automatic billing and so on. The present work aims to identify the benefits and implications of evolving the e-HUB architecture into a Software-as-a-Service (Saas) architecture towards the second commercial release of the e-HUB platform in support of SMEs competitiveness. Section two will describe the e-HUB [AS-IS] architecture, meanwhile section three will present the main concerns to design the e-HUB [TO-BE] Saas architecture. Finally, section 4 will depict a roadmap for the e-HUB evolution as well as the implications and benefits of having Saas requirements as the staring point for the new e-HUB architecture redesign.

### 2. THE e-HUB [AS-IS] SOFTWARE ARCHITECTURE

The e-HUB architecture was designed under an e-business vision rather than under a software functional requirements analysis. The e-HUB mission since its inception has been to stand for a new collaborative business infrastructure for SMEs. The e-HUB aim is to be a service provider of collaborative electronic solutions (e-services) for SMEs in order to support the creation and management of collaborative industrial networks among them as a new competitive business model.

The e-HUB software architecture design strategy, from a business perspective, was to lower the costs related to its implementation in order to provide a reasonable price for the e-services to the SMEs. This goal was reached thanks to an intelligent combination of open source technologies that made possible designing an affordable software architecture by integrating the following technologies:

- **Corporate portals** based-on a customized interface for each enterprise that allows a unified access point to its customers and suppliers besides the marketing features offered through a configurable content.
- **Enterprise Architecture Integration (EAI) and Service Oriented Architecture (SOA)** making extensive use of Web services to enable the e-services integration and interoperotion with external software applications.
- **Workflows and Business Process Management (BPM)** in order to maintain diverse processes within the enterprises business logic or the inter-enterprises business (B2B) logic.

Furthermore, the e-HUB physical architecture was designed as a multi-portal platform accessible through standard Web protocols (HTTP, WSDL), as an application container, and based-on a Java Enterprise platform (J2EE).

The e-HUB architecture offers thanks to its multi-portal platform a single access point to its tenants. This access point is represented by the SMEs Web portals for the deployment of all the e-services contracted by them.

Fig. 1 depicts the e-HUB architecture nowadays as a technological implementation of a robust platform that involves many components and technologies. The e-HUB architecture includes a set of components to deploy a number
of e-services and sub-portals for each SME. At top of the e-HUB architecture are the clients or consumers, these can be Web browsers and/or mobile devices with Web browsing capabilities, accessing with HTTP standard protocol or even with 3rd party applications the e-services through Web services technologies.

The main component within the e-HUB architecture is a corporate portal with a portlet container (Java Specification Request-168) in order to achieve the e-services hot-deployment behaviour, each e-service is a portlet-application. An e-service implementation could be defined as a composite component from a combination of user interfaces (views) and portlets (controllers) that interact with the services stack (model).

A business process engine is integrated into the e-HUB architecture to manage many workflows like quotation requests, work orders, purchase orders, etc. It interacts with other components through its Application Programming Interface (API) and the services stack provided by the portal platform it-self. The component that serves as a broker for the entire architecture is this services stack containing the definition and implementation of applications, database connections (JDBC pools), a content management system (CMS), security certificates (LDAP), an organisational management (roles, users, permissions), among others. In the lower part of the e-HUB architecture is the technological-physical infrastructure that supports the services stack, this layer consists in a set of collaboration services (SMTP, POP3), a file repository, network services (FTP, SFTP), security (Kerberos, SSL), and database management systems (PostgreSQL).

At infrastructure level, security concerns are supported by J2EE technologies such as JAAS (Java Authentication and Authorization Service) combined with physical certificates through Kerberos and SSL configurations. The security model used is the one embedded within the portal platform. The model is based-on roles and access levels. The combination of both security levels provides the e-HUB architecture with a robustness information protection and isolation.

<table>
<thead>
<tr>
<th>Technology</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>J2EE Web Container</td>
<td>Apache Tomcat</td>
</tr>
<tr>
<td>Portal Platform</td>
<td>eXo Platform</td>
</tr>
<tr>
<td>Web Services</td>
<td>Apache Axis</td>
</tr>
<tr>
<td>Business Processes</td>
<td>jBPM</td>
</tr>
<tr>
<td>MVC framework</td>
<td>Apache Struts</td>
</tr>
<tr>
<td>Database Management</td>
<td>PostgreSQL</td>
</tr>
</tbody>
</table>

The e-HUB services are Web applications deployed over the enterprise portals. Enterprise portals are containers that provide a single access point to every e-service. Furthermore, the e-services are presented to the user as portlets encapsulated in an enterprise portal.

3. THE e-HUB [TO-BE] SaaS ARCHITECTURE

3.1. The New e-HUB SaaS Architecture

New Internet-enabled platforms had appeared in recent years enabling open collaboration between enterprises. The e-HUB represents one of such platforms aiming to deliver software applications over the Internet under a SaaS approach. Based-on these new online models, the platform role has gone beyond developing and distributing software applications, and instead starting to manage servers and infrastructure to run e-services. In this new model the platform provider is also responsible for providing technical support. This set of software-infrastructure-support composes the whole service provider role.
In the customers’ case, software requirements are less strong since all they need is to have access to an Internet connection and an industry-standard browser or Web Services client. This online software delivery model is now conceived and defined as a SaaS model. SaaS is a well established phenomenon in some areas of enterprise IT and it is growing into a mainstream option for software-based solutions.

A SaaS platform can be defined as a set of software tools that enable the creation of applications with disregard of the delivery model requirements. The former means that a service provider (and developer) has not to worry about things like: data isolation, account management, billing model or any other thing unrelated to the application domain. Table 2 presents some characteristics or requirements for a SaaS platform.

The SaaS characteristics represent high level features that must be presented on a SaaS platform. Common SaaS platform implementations have to follow the next design attributes: (1) multitenant - supporting large number of tenants; (2) single version - referring to a single version per application and this one is available and shared among all the application clients; (3) logical data separation - with just one data domain for each tenant but all the data is stored in a single database schema; (4) multi-supplier - with the ability of sharing a single ICT-I among multiple SaaS vendors; and (5) application integration - being able to communicate each enterprise application with each others but they are deployed independently.

### Table 2. SaaS Platform Characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-Tenancy</td>
<td>SaaS platforms support the deploying and hosting of multiple application providers and clients.</td>
</tr>
<tr>
<td>Fine-grained Usage</td>
<td>SaaS application tenant pays the application providers in some proportion of usage.</td>
</tr>
<tr>
<td>Tracking &amp; Metrics</td>
<td>SaaS supports thousands of tenants with hundreds of thousands of customers each.</td>
</tr>
<tr>
<td>Extreme Transaction</td>
<td>SaaS style application ability to participate in composite applications.</td>
</tr>
<tr>
<td>Processing</td>
<td>SaaS ability to offer the same application on-premise, depending on the user requirements.</td>
</tr>
<tr>
<td>Integration with others Resources</td>
<td>SaaS applications can be localized per tenant or user in different geographies.</td>
</tr>
</tbody>
</table>

Fig. 2 depicts a general SaaS architecture, which at its top level shows a set of application services plugged-in as potential vendors. The SaaS platform exposes business services to the application services (e.g. application management, billing and metering services, and so on). Data services and tenant management provides data access and data customization for hosted tenant. Operating system and hardware components are the architecture baseline.

Service applications can be deployed and delivered for quite several customers without a significant increase in the required effort. A service application is a set of components that can be seen as a whole software application. Commonly an application could be composed by several modules which could be fired up in a per tenant basis.

### 4. THE e-HUB EVOLUTION TO A SaaS ARCHITECTURE

Integrating e-services through an e-HUB has advantages from separated and isolated Web applications. However, this could be improved with new available technologies as well as from the lessons learned through the PyME CREATIVA project. First it is necessary to assess the e-HUB platform and consider new ways of delivering the e-services. Lessons learned from the e-HUB architecture were:

1. **The business model should be painlessly integrated into the e-services instead of inserting ad-hoc code for supporting the business model in each e-service.** The supporting platform must provide mechanisms for implementing, adapting and managing the revenue model dynamically in a per service base and without affecting the service continuity. In this way, the service provider should define its own business rules through some interface provided by the platform.

2. **The subscription model must be e-service independent.** It is not recommendable to implement from scratch the subscription model for each set of e-services. The subscription model and its implementation must be managed by the platform not by the e-service.

3. **Monitor and log application usage is another functionality that should be encapsulated within the platform and must be based-on a functional usage monitor and a billing component in order to improve the business model support and accountability.**

4. **The platform must provide data isolation.** In the e-HUB approach, each application must implement with its own data source accesses such as database connections or Web services integration. This means that every e-service must be responsible for accessing only the data related to the current tenant. In an improved scenario, the platform must be responsible for making available the data to applications and assuring that the applications only gets the information from the current tenant.

5. **Multi-tenancy support and monitoring should be included in the platform in order to enable the definition of several tenant levels within an application context.** This would allow the creation of business models where any tenant could...
potentially become a service provider in a given application context. Multi-tenancy also represents the ability to create multiple providers with multiple applications. The platform should bring the ability to manage and monitor the applications per tenant.

(6) In order to improve the administration and implement a delegation model, the platform must provide means for delegating administrative responsibilities from the service provider to the tenant itself, by providing tenants with complete control over the platform provisioning aspects. Tenant and e-services administrators should have a control mechanism where they can manage users and roles, manage application subscriptions for tenant, view metering data and inspect provisioning problems. e-service’ subscribers must have control over their profile and their subscriptions.

(7) Implementing on-demand business process design and deployment. The ability of designing, deploying and managing business processes is a must for the next generation e-services platforms. Most middle and big enterprises are adopting this on-demand model because organisations must be able to run and change business processes in the long-term.

(8) Offering development and deployment tools and interfaces. A complete e-service platform must offer mechanisms and tools in order to design, develop, test, deploy and manage e-service applications. This would allow several vendors to become e-services providers by developing and managing their own applications and revenue models.

Lessons learned are pushing the e-HUB to a more service and subscription oriented infrastructure. It can be concluded that some components are missing from the actual e-HUB architecture. For instance, it is necessary a subscription component to support a dynamic subscription model, as well as a billing component to support automated, lights-out invoice creation. It is also possible to define new components such as metadata mechanisms or multi-tenancy support to free the e-services from all the logic unrelated to its core functionality.

Another missing component in the e-HUB is the mechanism to enable the development and hot-deployment of updated or new e-services to the platform. The actual version of the e-HUB supports a single provider developing and managing the e-services. With this implementation, it is very difficult for any other external entity (service provider) to develop new e-service applications. This happens because the current e-HUB platform lacks from an isolated environment to generate and deploy new applications (e-services). The availability of such tools will increase the e-HUB value offer by allowing 3rd parties to increase the number of e-services available at the e-HUB.

Recent trends in electronic services platforms points to SaaS platforms as solutions to the former needs. It is expected that the e-HUB and a pure SaaS platform share the majority of their components, since the e-HUB delivers its e-services using an early SaaS model. Transforming the current e-HUB architecture into a new SaaS implementation consists in redesigning the e-HUB architecture by adapting the e-services model to a pure SaaS model and integrating the application logic with the SaaS platform business services. In order to present how a typical SaaS solution fits within an improved e-HUB infrastructure, an assessment was performed, matching different SaaS platforms against the requirements presented in Table 3.

The assessment was helpful to understand why a SaaS offering could comply with the e-HUB requirements. Nevertheless, transitioning from the current architecture to a SaaS implementation is not a trivial effort. In the next paragraphs the main issues for transitioning from an e-HUB environment to a SaaS one will be presented.

From a single business model to dynamic subscription model. Current e-HUB implementation and most Application Service Providers (ASPs) define a single static revenue models (e.g. embedded & hardcoded within the application implementation). Revenue models can be based-on the number of users, periods of time, resources used or any combinations of the former. Unfortunately revenue models can not remain static. When the dynamic nature of markets asks ASPs to modify their revenue model, ASPs are not able to change it in cost-effective way, mainly because the revenue model is hard coded into the application.

<table>
<thead>
<tr>
<th>Platform/Requirement</th>
<th>Business Model</th>
<th>Multi-Tenant Support</th>
<th>Customization</th>
<th>Business Process Oriented</th>
<th>API for Development</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salesforce</td>
<td>Subscription for Year or User</td>
<td>Strong</td>
<td>Strong</td>
<td>Medium</td>
<td>Force.com</td>
</tr>
<tr>
<td>NetSuite</td>
<td>Subscription</td>
<td>Medium</td>
<td>Medium</td>
<td>Strong</td>
<td>NS-BOS</td>
</tr>
<tr>
<td>OpSource</td>
<td>Pay-as-you-Grow</td>
<td>Strong</td>
<td>Medium</td>
<td>Medium</td>
<td>On Demand</td>
</tr>
<tr>
<td>Coghead</td>
<td>Pay-as-you-Go Subscription</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
<td>Appian Anywhere</td>
</tr>
<tr>
<td>Appian</td>
<td>Subscription</td>
<td>Medium</td>
<td>Strong</td>
<td>Strong</td>
<td>Appian Anywhere</td>
</tr>
<tr>
<td>Servoy</td>
<td>Licenses</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
<td>Servoy developer</td>
</tr>
<tr>
<td>Cordys</td>
<td>Subscription</td>
<td>Strong</td>
<td>Strong</td>
<td>Strong</td>
<td>SaaS Deployment Framework</td>
</tr>
</tbody>
</table>
Imagine changing an e-service fee based-on time with no usage limits for one involving charges based-on usage, seats, bandwidth or anything that the business demand. Such change would require recoding the e-service to comply with the new requirements of information. This of course has costs associated.

Moving from a static business model to a dynamic one involves the definition of business rules and subscription models through the SaaS platform which in turn will affect the configuration of the whole set of e-services with a minimal cost associated.

From portal-oriented to provider-oriented.

Traditional ASPs provide a portal mechanism for accessing their applications. In the case of the e-HUB, multiple portals are deployed, one for each SME. This approach need to be changed in order to achieve a service provider-subscriber environment. The current implementation of the e-HUB only support the notion of one service provider, this is the host platform it-self.

The benefits of shifting to a multi-provider approach include an easy integration with associates that complement the platform administrator. Migrating to a multiple provider with multiple e-services also provides the ability to deploy and manage independent sets of applications. If every set of applications can be managed by a different service provider then the offering of e-services will be enriched by the capabilities or experiences of the whole set of service providers.

From Web applications to SaaS applications.

Actual e-HUB services are designed, developed and deployed as Web applications. They are managed by the platform through a Web container and there is no other support for them (such as versioning, back-up, restore, monitoring). In the other hand, we could see SaaS applications as desktop applications running within an operating system.

In this case, the operating system role is played by the host SaaS platform. The SaaS platform offers support for its hosted applications, such as back-up, restore or metering.

5. CONCLUSIONS AND FURTHER WORK

The e-HUB by its-elf is a strong technological platform that has to be changed not because technological reasons but because changes in the business landscape. The e-HUB has proven to bring benefits into SMEs. The changes required in the platform are mainly efforts to separate the business concerns from the e-services’ concerns in order to comply with changing market needs.
Evolving the e-HUB doesn’t mean a hard separation between e-services and administrative applications, mainly because there are relationships between what is going on in the domain of the e-services and the revenue model. Such relationships persist during their lifetime. In the traditional software delivery model there are no meaningful business concerns in the application domain once the software is in the hands of the client, but in the new SaaS model the software is never fully in customer’s hands, instead the service provider is continuously providing a service which in turn needs to be administered and monitored.

SaaS platforms are meant specifically for providing support to most of the aforementioned new requirements of the e-HUB, but it doesn’t mean that they are easy to adopt. Applications need to be created specifically to be mounted in top of those platforms. Application design must be focused on the functionality exposed to the client rather than supporting a specific business/revenue model.

It is clear that there would be benefits if the e-HUB is evolved into a new platform where e-services are easily deployed and managed by multiple providers, but there is still a lot of effort to effectively achieving the desired results and to replicate the success of the already implemented e-services. Further work will be done by migrating the current e-HUB to a SaaS environment in order to take advance of the benefits presented in this work.

ACKNOWLEDGEMENT

The research presented in this document is a contribution for the “Rapid Product Realization for Developing Markets Using Emerging Technologies” Research Chair, ITESM, Campus Monterrey, and for the “Technological Innovation” Research Chair, ITESM, Campus Ciudad de México.

REFERENCES


SHORT-BIOGRAPHIES

David Concha is a Researcher and Programmer at Center of Innovation in Design and Technology of the Tecnológico de Monterrey, Campus Monterrey since April 2004. From this institute, he also received his Master’s Degree in Information Technologies (2006). As part of the research center, he has worked in different national and international projects related to on-demand databases, business process management and IT-platforms development. He was an invited researcher to the Hewlett-Packard (HP) Lab at Palo Alto, California from April 2007 to April 2008.

Javier Espadas is a PhD candidate of the Information and Communication Technologies Programme at the Tecnológico de Monterrey, Campus Monterrey. He is also a Researcher and Programmer at Center of Innovation in Design and Technology of the same institute since April 2004. As part of the research center, he has worked in different national and international projects regarding the development of IT-platforms based-on state-of-the-art approaches such as service-oriented architectures (SOA) and software-as-a-service (SaaS) models in collaborative environments.

David Romero is a Sr. Research Scientist and Project Manager at Center of Innovation in Design and Technology of the Tecnológico de Monterrey, Campus Monterrey and Mexico City since June 2005. From this institute, he also received his Master’s Degree in Administration of Information Technologies (2007). As part of the research center, he has participated in various national and international projects related to enterprise integration, interoperability & networking and collaborative networked organisations.

Professor Dr. Arturo Molina is the General Director of the Tecnológico de Monterrey, Campus Mexico City. He received his PhD degree in Manufacturing Engineering at Loughborough University of Technology, England (1995), his University Doctor Degree in Mechanical Engineering at the Technical University of Budapest, Hungary (1992), and his Master’s Degree in Computer Science from Tecnológico de Monterrey, Campus Monterrey, Mexico (1992). Professor Molina is member of the National Researchers System of Mexico (SNI-Level II), Mexican Academy of Sciences, and member of IFAC TC-WG 5.3 on Enterprise Integration and Enterprise Networking, IFIP WG 5.12 on Enterprise Integration Architectures and IFIP WG 5.3 Cooperation of Virtual Enterprises and Virtual Organisations.