Normalization of Industrial Machinery with Embedded Devices and SOA

Virgilio Gilart-Iglesias,
Francisco Maciá-Pérez
Computer Science Department
University of Alicante
PO Box 99, 03690, Alicante, Spain
{vgilart, pmacia}@dtic.ua.es

Francisco José Mora-Gimeno,
José Vicente Berná-Martínez
Computer Science Department
University of Alicante
PO Box 99, 03690, Alicante, Spain
{fjmora, jvberna}@dtic.ua.es

Abstract

In the present paper we propose a method that permits visualization of manufacturing devices from a functional perspective. The aim of this method is to raise the abstraction level of manufacturing devices from the lower levels of production to the enterprise level of the business model. The target of this new approach is to integrate in a transparent way the resources, processes and, in general, the business logic of manufacturing levels using existing business models and achieving business continuity and process automation. The proposed method comprises two phases: the first phase that we have named industrial machinery normalization process*;* and the second phase that we have named *manufacturing processes reorganization process. This paper focuses mainly on the former. ie specification of the hardware normalization and industrial machinery functionalities process.*

1. Introduction

Internet has given customers throughout the World the ability to choose the consumer goods that best suit their needs, and at the lowest possible prices. This fact has driven the evolution of industry, from traditional manufacturing paradigms, towards new models in order to facilitate massive customization [2].

Customers are no longer an external entity outside the manufacturing process; instead, they have become an active part of this process, by determining the specific features that each product must have.

In order to achieve this goal, a true integration of the supply channels (SCM) is required between the implied organizations: manufacturers, suppliers, auxiliary and logistical enterprises must work in a perfectly synchronized fashion, continuously taking into account every stage in each manufacturing process, independently of who is responsible for its execution, where it is to be performed and, specially, the location of the information required in order to complete it [8].

One of the main causes that prevent a true integration of the supply channels is that the infrastructures at the manufacturing levels do not use the new technologies' full potential, especially at the organization's lowest levels, where the production elements are located (PLC, CNC, industrial machinery, etc.).

Obviously, such scenario does not allow deploying abstraction tiers in order to upgrade the organization's lowest levels until they reach integration with the higher levels, therefore becoming part of the business model.

Despite these facts, mature technologies arisen with the Internet —n-tier architectures, both B2C and B2B, middleware-based applications and service oriented architectures— have been introduced in other business scopes such as financial management, resource planning (ERP), customer service (CRM) or on-line sales (eCommerce), thus allowing to overcome traditional barriers that prevented business strategies to be aligned with the customers' likes and needs [8] [19].

However, eBusiness concept's success is strongly tied to the integral handling of every business scope [1], and therefore the SCM's integration problems have deeper implications than what could be expected at first sight, preventing investments and efforts in other business areas from being fully capitalized.

In eBusiness models, the concept of service plays a significant role —with emerging terms like "Software as a Service" (SaaS) or on-demand applications—, in the present paper we outline an extension of this concept as "Industrial Machine as a Service" (IMaaS). In order to do so, we must first meet the technological requirements of the production elements, which will allow us to suppress the current physical barriers that prevent them from reaching the necessary abstraction level.

We can achieve this by introducing embedded systems with normalized interfaces at our architecture's different layers. Communication networks like Ethernet or WiFi, network and application protocols such as TCP/IP, SSL, HTTP, SOAP, and UDDI will play a main role throughout this process. Finally, we will have a platform suitable for conceptually presenting the manufacturing elements as services. This process is our proposal's foundation, and we have named it as industrial machinery normalization.

In the following sections we will analyze the current research works related to the production levels' integration in the electronic business models, and the most relevant advances and applications of embedded systems.

Next, we will expose the global framework in which our proposal is located, and we will describe the industrial machinery normalization process, which will allow us to upgrade its functionality at the same level as the rest of the organization's business logic. In the fourth point, we will propose a development scenario in which to perform the tests required in order to validate our proposal. Finally, we will present the main conclusions derived from our work, together with the current research lines.

2. Background

Internet's evolution has caused the adoption of new strategies by organizations in order to adapt their processes —process reengineering— [1], and the use of enterprise paradigms and architectures based on distributed software components on n-tiers, which allow organizations to introduce new business models and to take advantage of the new competition model [18].

Software components encapsulate the business logic and provide a nimble tool in order to adapt the enterprise's objectives and strategy to the changing environment. However, due to physical and technological constraints, manufacturing processes have not yet reached the desirable integration level, and in fact they are regarded as inherited systems in most cases. In [3], the author gathers every communication and integration technology currently used at the manufacturing levels as external systems outside the business processes —for instance: Modbus, Profibus, AS-I, FIPIO, DeviceNET, Interbus or industrial Ethernet. This work is centred in the traditional model of industrial automation based on proprietary protocols which require ad-hoc adapters placed in the resources level of the eBusiness model, in order to integrate with the business components located at the enterprise level.

Schneider has been one of the first manufacturers of automation and industrial control devices who has proposed the introduction of embedded devices and Internet paradigms —Ethernet, TCP/IP and Web protocols— in its automatons, in order to enable them to communicate with the management applications. This tendency can be seen reflected in concepts such as transparent factory [7]. In [9] several researchers from ABB enterprise propose the introduction of embedded systems in the control devices on top of which rely widely spread Internet protocols, like SOAP, so as to establish communications with the higher levels. In this work the control device is endowed with *intelligence* and self-management capabilities, not only setting an interface that provides access to its functionalities, but also proactive capabilities that allow it to initiate on its own communications with the management systems, in the face of certain events.

In [6], the author proposes to use Web Services as the interface that provides access to the functionalities in control and automation devices, in order to facilitate its integration with the enterprise resource planning systems (ERP).

These three proposals are centred in the automation and control levels' elements, keeping the currently used technologies in order to communicate with industrial machinery. They abstract such devices as manufacturing processes but do not define their location together with the enterprise business processes, inside the eBusiness model's general map.

Inside the ITEA [13] initiative's frame of European research and development projects, the SIRENA [12] project is under development, with the objective of creating a framework for the specification and development of distributed applications on real-time embedded systems, such as industrial automation and automobile industry. As a result of this project, in [10] [11] the author presents an approach based on SOA architectures and the provision of infrastructures in embedded network devices in general, and particularly in industrial machinery, thus enabling them to be presented as services. In [14], the author sets a proposal in order to coordinate these new elements as manufacturing processes in order to get a higher level process.

However, in our research project, the proposed approach is centred on getting an integral business model which allows business continuity (from customer order to product manufacturing and delivering). In this way, the organization's global business process is optimized. To this end, we propose a procedure that we have called 'normalization process'. This normalization process follows a methodology similar to the SIRENA project's one [12], but aiming to a different target. Our goal is to achieve higher abstraction levels in manufacturing devices, in order to incorporate enterprise class features (n-tier architectures and distributed software components) at the production environments.

3. Proposed approach

The work described in this paper is included in a more extensive research whose main target is to achieve an integral eBusiness model, compatible with the currents proposals, to overcome the physical and logistical constraints, which prevent processes from taking advantage of the capabilities provided by new technologies at this enterprise level. This potential is already being used in other fields.

The method followed in order to reach our goal consists in dividing the process in two separate steps, after having analyzed the current integral eBusiness models. The first step, which we name as production device normalization process, is aimed to manage the

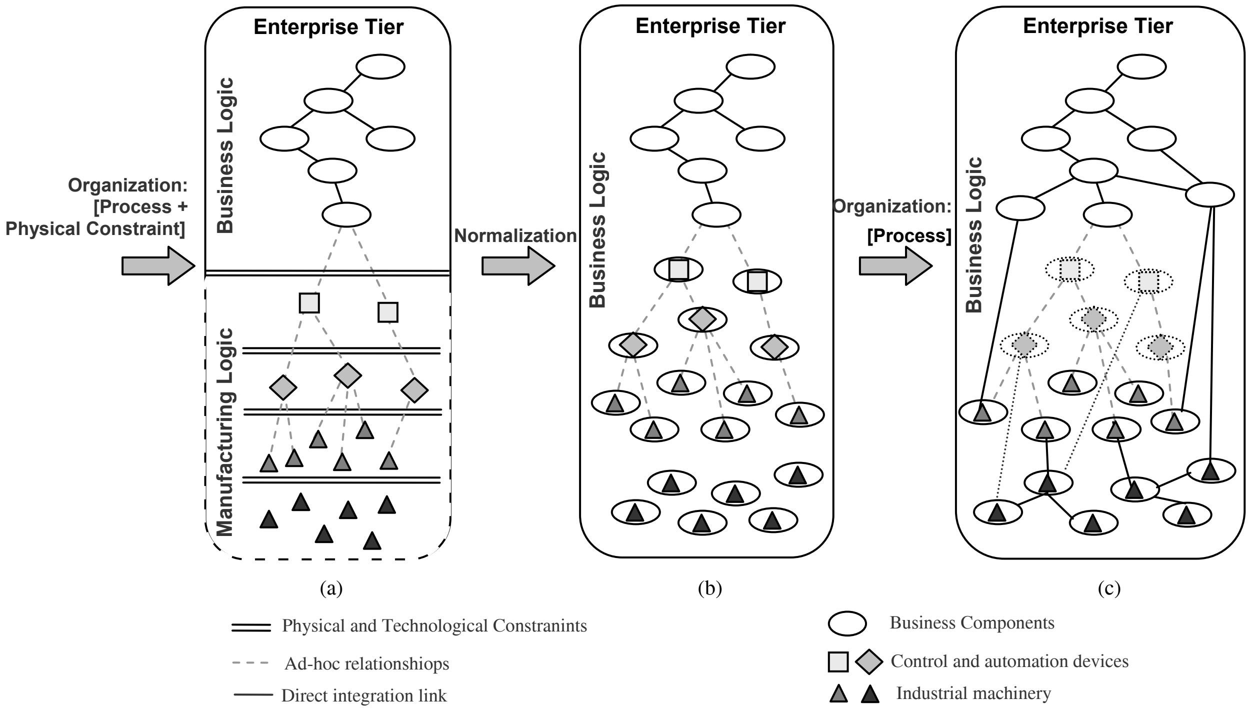

Figure 1. EBusiness proposed approach.

industrial machinery and its functionality at the same abstraction level as the rest of the business logic; once these elements are normalized, we include them in the eBusiness model, so that they are literally merged into business logic.

The figure 1 presents, in a schematic way, the main elements involved in the business processes, which belong to a manufacturing enterprise. In each option —a, b and c— we show how the relationships between these elements change, as the process goes forward.

The figure 1.a shows a typical scenario nowadays, which is the result of the organization work based on processes which affected companies during the 90's, being influenced by the physical and technological constraints of the industrial machinery. These constraints were caused by proprietary rules —communication interfaces and protocols. Due to these restraints, the business logic was divided in two big blocks, dedicated to the business logic and the manufacturing logic. In this diagram, the control and automation physical devices, characteristic of this period, have been introduced inside the manufacturing section. They have been used as a technological bridge, by means of ad-hoc adapters, in order to resolve the integration issue.

At the figure 1.b, the manufacturing devices (control and automation devices, industrial machinery…) have been already normalized (this is the main target of this paper and it is described in the next section). This industrial machinery normalization process places the manufacturing devices (PLC, CNC, sensors, actuators, etc.) at the same level as the business components by means of embedded devices and the ICT technologies arisen with the Internet. Therefore, the differences between both sections' components are suppressed (physical and technological constraint), allowing us to apply high level patterns —as in the business logic. Although, being realistic, it is not possible to suppress the inherited control and automation systems at the beginning. These inherited systems have been resolving for a long time, in a better or worse way, these technological differences between the business logic and the industrial machinery. For this reason, we consider this intermediate phase, in which the inherited relationships are maintained, with the advantage that there are no physical or conceptual constrains, but only procedural restrictions, and from now on these relationships can be established by mechanisms that facilitate their management, such as on-demand services and service oriented architectures.

The figure 1.c shows the desired scenario, which is the final target of the project where the present job is included. First, the production devices which were initially isolated from the systematic relationships have been integrated in a natural way. Next, the intermediate elements will be progressively suppressed and the final relationships will be established between all of the components, which will be guided solely by the procedural organization.

Currently, we are working on the specification of the new relations between processes and their organization in order to achieve an integral and continuous eBusiness model. We have named this procedure as *manufacturing processes reorganization.* In this stage, our efforts are focused on the research of the standards for business enterprise centred on process management (ISO norms).

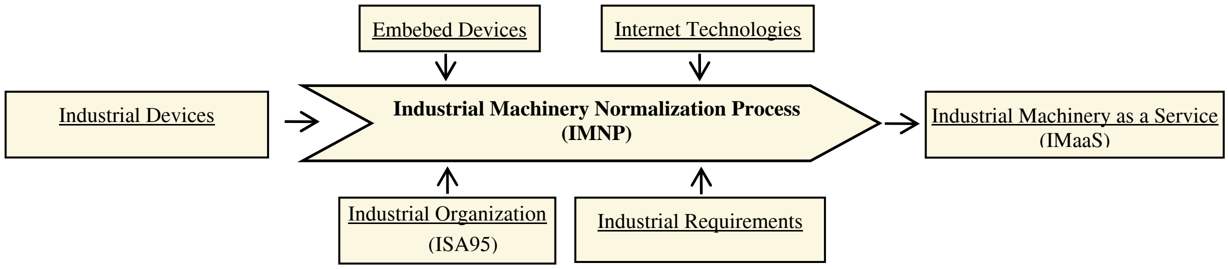

Figure 2. Industrial machinery normalization process model.

4. Industrial machinery normalization process

The objective of the normalization process is to characterize the production elements —including the industrial machinery—, from the point of view of its contribution to the company's business model. In this way, a new concept allows the evolution from technical elements —industrial machinery and ICT infrastructure— to IT services in first place, and finally to business processes.

In order to reach this objective, the abstract tiers of the manufacturing elements must be upgraded so as to level them with the company's ICT infrastructures, which in this case is the software container for a n-tier architecture. Once the normalization of the physical components is finished, we will be ready to do the same with their functionality. In this case, the process will express the manufacture process in terms of distributed software components and service, executing them inside embedded component containers and therefore integrating them at the enterprise level, inside the company's general business logic, and establishing a unified scope where all the business processes of the organization are placed.

This approach provides continuity to the model and the development of the business process, including all the enterprise levels, thus allowing the application of industry standard models oriented to process management (ISO).

Therefore, this process is developed in two stages: first, specifying the requirements and establishing the physical architecture that enables the production devices to be transformed in an embedded software container and, second, defining the embedded component model and service software so as to encapsulate the devices' functionality. Figure 2 represents the normalization process of the industrial machinery in which the following elements are present:

- Industrial devices. These consist of the basic input elements which will be transformed by the normalization process. They comprise all the elements of production which are involved in the control, automation and process levels: PLC, CNC, sensors, actuators, industrial machinery etc.
- Industrial requirements. During the normalization process the restrictive characteristics of the industrial environment should be taken into account: real time, security, heterogeneity, physical restrictions etc.
- Industrial Organization. The result of this normalization process should be ISA95, PERA Model, etc. In fact, the normalization process is not required to reorganize new industrial elements. This is the task of subsequent phases of the proposed methodology.
- Embedded devices. These are the (basically electronic) elements which provide industrial devices with the minimum *intelligence* on which to develop the proposal. This minimum intelligence may be specified as: computational capacity, memory and communication.
- Internet Technologies. The purpose of the normalization process is to raise the level of the industrial elements to the same levels of abstraction achieved by the other business components, so that in the second phase they will be integrated in a transparent manner in a global eBusiness model. For this reason we will need to apply the same technologies, standards and paradigms which we have already successfully used in higher levels of the business. Therefore, we shall define a container for embedded software components over the platform provided by the embedded systems, (based on high level Internet protocols: such as HTTP, SOAP or UDDI) which facilitates the introduction of service orientated architectures.
- Industrial Machine as a Service (IMaaS): These elements are the result of the transformation of industrial devices. At this point industrial elements have reached the same level of abstraction as the business logic at a enterprise level, and may be considered from the perspective of their functionality and, although this is not the initial phase objective, they are actually ready to be integrated in a transparent manner into the n-tier architectures and interaction models (B2C, B2B, M2M) of the enterprise as software components which encapsulate the business logic defining their functionality.

4.1. Physical design

The first step of this phase consists in endowing the production devices with computing and communication capabilities. In the last years, advances in electronics and communications have given birth to a new category of small size, low cost computers, therefore providing enough processing power for our target. These computers are the so-called embedded devices, which enable the integration of other systems, providing them with the required features —communications and computing power— [4] [5]. By using these devices, we can provide advanced functionalities and distributed

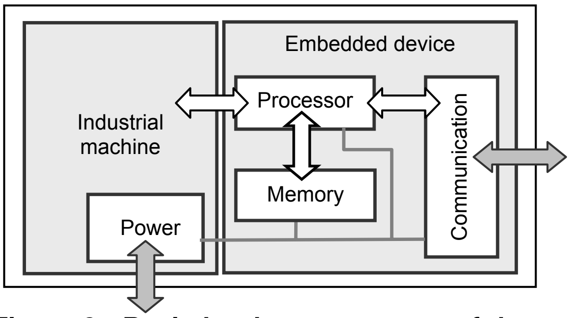

Figure 3. Basic hardware structure of the production devices.

computation paradigms to the production elements. Such functionalities and paradigms have been widely used as the Internet has evolved, and finally they have led to a new generation of smart devices [15]. Figure 3 outlines the general structure of these new smart production

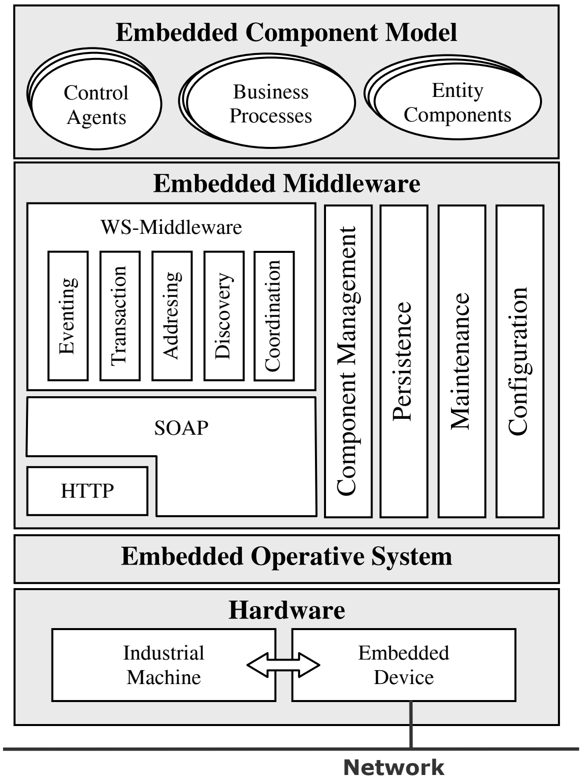

Figure 4. Software architecture of the production devices.

devices, whose hardware architecture is integrated by industry machinery and embedded devices.

In our research fields, we do not cover the design of the embedded devices, but use them solely in order to achieve the normalization. The majority of these devices provide high level Internet protocols such as HTTP which will be used as a basis for constructing the embedded container. The second step defines the embedded component and service container which provides the infrastructure, in terms of middleware services, used by the software components, which encapsulate the functionalities of the different industrial devices (figure 4).

In this way, the system functionality will be determined by the software and not by the hardware, therefore providing a higher flexibility, autonomy and interoperability to the production devices [4].

Production elements are thus defined in terms of software containers and software components, similarly as the way Internet applications are defined in terms of application servers and Web containers. This endows the production components with a set of services which are appropriate for the market's social and technological tendencies: integration, self-management, zero configuration, autonomy, hardware and software platform independence, flexibility, security, fault tolerance, business continuity and ease of use.

4.2. Component model

Once the software container is established as the execution framework for the software components, a profiler is used in order to characterize the different categories in the application's functionality (figure 4). This approach distinguishes three different types of software components in the application layer at the manufacturing elements: business processes, control agents and unit components.

First, business processes implement independent work entities that encapsulate the business logic for functionalities provided by the mechanical elements. Endowed with a passive behaviour, they are continuously expecting requests from external components, following the service-oriented paradigm — in which they act as servers for the functionality they provide. These components are analogue to their homonym ones at the enterprise level of the usual eBusiness model.

Second, control agents define the active, smart behaviour of the production elements. These components communicate in an autonomous way with the external control units, sending them information in order to verify the correct use of the manufacturing elements as the production process is carried out, therefore providing an added value to the company's global business (Operation error warnings, industrial machinery reconfiguration, maintenance control, etc.).

Ultimately, the entity components encapsulate the necessary information, for both customizing the business logic represented by business process components and providing the control agents with the status variables which will guide their operation (for example time ranges of an operation indicating the need for maintenance or configuration values of the device in order to adapt to changes in the production line.

This model provides production elements with the flexibility required to adapt to production changes and to improve continuity of the business organization.

5. Use case

As an example of the normalization process' implementation, we have defined a scenario in order to carry out a simple use case, which has allowed us to obtain the first results in this research. The work has been divided in the following tasks: outline of the general scenario, specification of the functional design, components description and implementation.

5.1. Outline of the scenario.

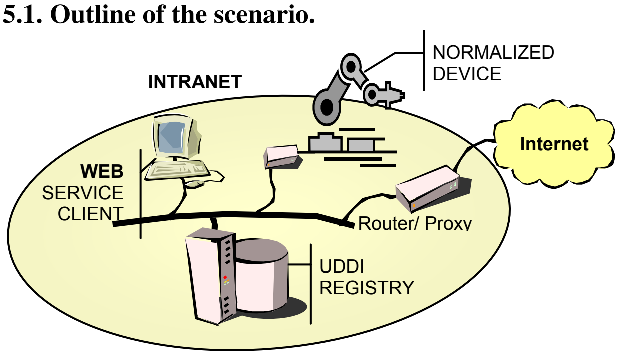

Figure 5. The proposed scenario.

The proposed scenario (figure 5) consists in three elements connected through a network: a robotic arm to which the normalized process has been applied, introducing an embedded device, an UDDI registry server and a client computer. After its normalization, the robot arm is transformed in a Web Service that is registered in the UDDI server. From the client computer we can access the UDDI registry in order to obtain the WSDL document that describes the service so as to establish a user interface that provides access to the industrial device's functionality.

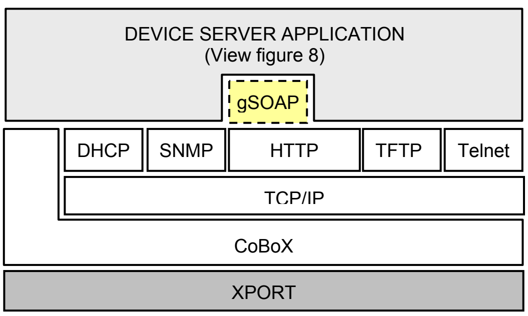

Figure 6. XPORT software architecture.

5.2. Components description

In order to carry out our experimentation, we have used the following components: a manipulating robotic arm model RTX by OxIM, an XPORT embedded system from Lantronix [16], a server including the UDDI registry and publishing service for Intranets (provided by Windows 2003 server) and a client computer with the Visual Studio .NET 2003 integrated development environment.

The RTX is a manipulating robotic arm with 6 degrees of freedom, placed on top of two guiding rails which provide an extra horizontal movement freedom of 2 meters.

The robotic system can be controlled by issuing ASCII command sequences through a serial port 32-pin RS-232 interface. These commands allow us to set each articulation's motor, the angle and speed of each movement, the hand-grip's aperture angle and also to return to the initial position.

The XPORT is a small-sized embedded device (its dimensions are 17x14x34 mm), endowed with an RJ45 female connector which enables it to link into an Ethernet 10/100 BaseT network. It also has a 256 KB SRAM memory and a 512 KB Flash memory. Its 16-bit 88 Mhz microprocessor is compatible with the 8086 architecture. Besides, it features a serial interface which supports the RS-232, RS-422 and RS-485 standards. Its software architecture is described in figure 6. The SOAP transport protocol, fundamental for our objective, is not included in the XPORT. In the implementation process we describe how we resolved this lack.

5.3. RTX normalization process

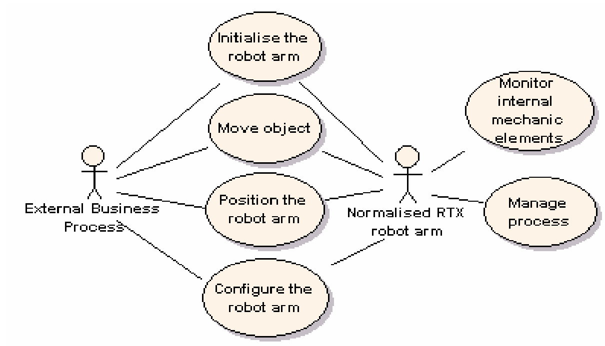

Figure 7. RTX robot arm use case model.

In order to carry out the experiment, a subset of functionalities of the robot arm was specified (figure 7). These functionalities were divided into two groups. The first are concerned with the functionalities to be presented as Web Services to the external business processes. These functionalities will be encapsulated in the business process components (view figure 4). The second are related to the autonomous behaviour of the robot in order to ensure maintenance and continuity of the operations. These functionalities are represented by control agents (view figure 4):

In order to normalize the robot, we have defined in addition, a business process component (RTXBusinessProcess) which will provide the following functionalities:

- Initialise the robot arm: The robot arm returns to the initial position.
- Move object: Move an object from the original position X and the final position Y.
- Position the robot arm: Move the arm to the position indicated.
- Configure the robot arm: Define the highest level functionality. The robot arm would be able to handle elements of various dimensions in a production line. When the production process changes an element, it is necessary to readjust the robot parameters in order to handle this element correctly (degree of aperture of the gripper based on the dimensions of the element to be handled, initial position of the element and final position, number of elements to be handled in a production batch etc). This is an extremely useful option for ensuring continuity and flexible automation in the production process of an organization.

In addition we have defined two control agents:

- Process agent (manage process): This agent establishes alarms for any continuity problems in operations entrusted to the robot arm (for example availability of elements to be manipulated, errors in the dimensions of the element to be handles etc.).
- Maintenance agent (monitor internal mechanic elements): This agent controls the correct operation of the mechanics and the robot components monitoring work and movement times, comparing them to a range of values which define *adequate operation (*Speed of movement, motor function etc.)

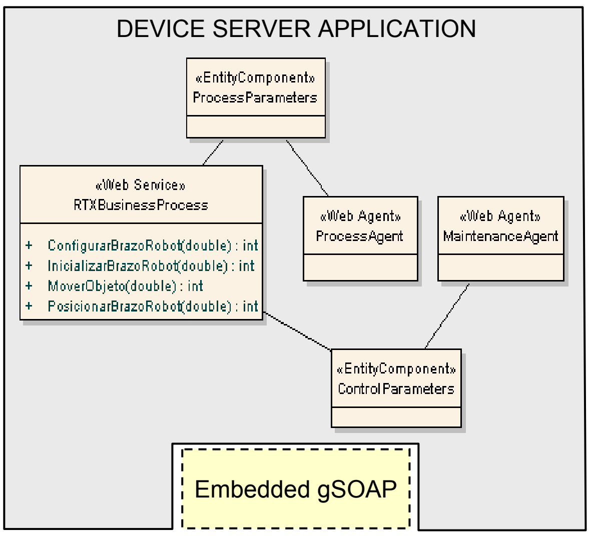

Figure 8. Device server application model.

in order to establish preventive maintenance.

The XPORT has a monolithic architecture with a single execution process divided into different applications which make up the collaborative multitask system (change of task is carried out in an explicit manner). The device software comprises a programme written in C language, where each task is a function registered in the collaborative scheduler. In order to implement the aforementioned use (figure 8) we have defined three collaborative tasks (each one of which represents the components defined in the functionalities of the robot arm) which are implemented by three functions registered in the main function of the device firmware (device server application). These tasks are sustained over a container which implements the basic services required to provide SOA architecture.

5.4. Use case implementation and deployment

The implementation of the outlined use case has been carried out a priori in a PC with the required development tools: TurboC for the two XPORT components and VS.NET for generating the client interface that uses the service; and the gSOAP library [17], which implements the SOAP protocol in the C programming language.

After generating and validating the application that encapsulates the robotic arm's manipulation business logic, we created the infrastructure required in order to provide our application as a Web Service. We generated the WSDL document describing our Web Service. After this step, the document was registered within the UDDI service included by Windows 2003 server. Next, we used the VS.Net development environment together with the WSDL document so as to generate a client interface to access our Web Service. This step was easily conducted, and access to the Web Service was achieved as if we were accessing a local component. Next, we implemented a simple user interface so that the final user could access the robot's functionality, therefore enabling our application to be validated. When we obtained successful results, we deployed our application into the XPORT device. In order to do so, it was necessary to adapt the gSOAP library to the embedded device, due to the memory constraints, and also to transfer the application to the XPORT via TFTP. We connected the XPORT embedded device to the robotic arm through an RS232 to RS485 adapter, and then linked the XPORT to the Ethernet network through its RJ45 port. The client application was updated with the new IP address required in order to locate the Web Service.

6. Conclusions

In this work we have presented a proposal aimed to conceptually integrate the production elements —basically, the manufacturing machinery— inside the eBusiness' global model, thus suppressing the traditional dependency of this machinery's physical and operative features.

The proposed method comes from the normalization concept, by means of which we seek the characterization of the production elements from the point of view of its contribution to the business model. Once normalized, they can be integrated in a transparent manner inside the conceptual eBusiness model.

The fulfilment of our proposal is deeply favoured by the current state of the embedded devices technology, which allows us to embed inside the industrial machinery the computing hardware, communication protocols, service layers and *intelligence* required in order to achieve our goal.

We are currently working in the second phase of our proposal, placing the new business components arisen from the normalization process inside the technical architecture, both conceptual and physical, of the eBusiness model.

References

[1] P. Harmon, M. Rosen, M. Guttman. *Developing E-business Systems and Architectures: A Manager's Guide.* Morgan Kaufmann Publishers, San Francisco, USA. 2001.

[2] C. Younghwan, K. Kwangsoo, K. Cheolhan. "A design chain collaboration framework using reference models," *International Journal of Advanced Manufacturing Technology*. 26 (1) pp. 183-190. July, 2005.

[3] R.P. Moreno. *Ingeniería de la automatización industrial*. Ra-Ma, Madrid, Spain, 2004.

[4] J. Catsoulis. *Designing embedded hardware. Second edition*. O'Reilly, USA, 2005.

[5] D. Gill, J. Hobday. *Internet embedded devices for industrial applications*. ERA Technology, UK. 1999.

[6] A.P. Kalogeras, J.V. Gialelis, C.E. Alexakos, M.J. Georgoudakis, S.A. Koubias. "Vertical integration of enterprise industrial systems utilizing Web Service," *Proceedings of the 5th IEEE International Workshop on Factory Communication System* (WFCS 2004), Technical University of Vienna, Vienna, Austria, 22-24 September 2004.

[7] Transparent Factory. *Manual de usuario y planificación*. [Online] Available: http://www.modicon.com, 2001.

[8] S.L.S. Worthington, W. Boyes. *E-Business in Manufacturing: Putting the Internet to work in the industrial enterprise*. ISA Press, 2002.

[9] U. Topp, P. Müller. "Web based service for embedded devices," *International Workshop on Web Service: Research, Standardization and Deployment* (WS-RSD'02). Lecture notes in computer science. Web, Web Service and Database Systems, pp. 141-153, 2002.

[10] F. Jammes, H. Smit. "Service-Oriented paradigms in industrial automation," *IEEE Transaction on industrial informatics*. VOL I. nº 1, pp. 62-70. 2005.

[11] F. Jammes, H. Smit. "Service-Oriented architectures for devices –the SIRENA view," *Proceedings of 3rd International Conference on Industrial Informatics*. INDIN 2005.

[12] The SIRENA project. [Online] Available: http://www.sirena-itea.org

[13] The ITEA initiative. [Online] Available: http://www.itea-office.org

[14] F. Jammes, H. Smit, J.L. Martinez-Lastra, I.M. Delamer. "Orchestration of Service-Oriented Manufacturing Processes," *Proc. of the 10th IEEE International Conference on Emerging Technologies and Factory Automation,* ETFA 2005, Catania, Italy, September 19-22, 2005

[15] V. Gilart-Iglesias, H. Ramos-Morillo, F. Maciá-Pérez. *Del eBusiness al Smartscrew. Servicios electrónicos para la sociedad de la información: Desarrollo de grandes aplicaciones distribuidas sobre Internet*. JDARE 2005. ISBN: 84-7908-850-8. 200.

[16] XPORT device. [Online] Available: http://www.lantronix.com/device-networking/embedded-device-servers/xport.html

[17] GSOAP Library. [Online] Available: http://gsoap2.sourceforge.net/

[18] F. Maciá-Pérez, V. Gilart-Iglesias, J.M. García-Chamizo, A. Hernández-Sáez, D. Marcos-Jorquera. "Decoupling MVC: J2EE design patterns integration," *Proceedings of the 7th International Conference on Enterprise Information Systems*, ICEIS'05. INTICC Press, 2005.

[19] V. Gilart-Iglesias, F. Maciá-Pérez, J.M. García-Chamizo, A. Hernández-Sáez, D. Marcos-Jorquera. "A model for developing J2EE applications based on design patterns," *Proceedings of IADIS International Conferences on Applied Computing*, IADIS-AC'05. IADIS Press, 2005.