Advanced Enterprise Process Modelling Utilizing Ontology Semantics

J. Gialelis¹, A.P. Kalogerias², C. Alexakos², M. Georgoudakis¹ and S. Koubias¹

¹Applied Electronics Laboratory
Electrical and Computer Engineering Dept
University of Patras Campus
26500 Rio Patras, Greece
{gialelis, georgoud, koubias}@ee.upatras.gr

²Industrial Systems Institute
Science Park of Patras Building
Stadiou Str, 26506 Platani Patras, Greece
{kalogeras, alexakos}@isi.gr

Abstract

Ontologies present a state-of-the-art technology for the provision of machine-readable semantics. Its utilization for the annotation of end-to-end enterprise / industrial processes characterizing advanced collaborative business models, largely contributes to enterprise interoperability, flexibility and component reusability. Our work in this paper emphasizes on the use of ontologies in combination with workflows for business process modeling and presents an associated methodology and architecture.

1. Introduction

Today’s manufacturing enterprises must be agile and responsive in order to be competitive. Furthermore, the penetration of information technology (IT) into almost every aspect of operations is transforming business environments from production-centric to information and customer centric. The rapid progression towards a “mass customization” business model, where products are sold based on a build-to-order basis, has significantly altered the traditional definition of the supply chain by adding a dynamic entity, the customer, into the process. Such dynamic entities cause changes in inter and intra organizational environment. Therefore, industries should flexibly react to changes in markets and trading partners and to successfully deal with required internal technical and organizational changes.

One of the major changes has taken place in the way enterprises cooperate with each other by forming manufacturing networks in order to optimally collaborate towards the production of products or the provision of services. Such networks, which aim at greatly enhancing manufacturing functionalities, flexibility and effectiveness are defined as collaborative manufacturing networks. Thus, collaboration among enterprises is increasing throughout the entire product life cycle.

In order to satisfy the above requirements, collaborative manufacturing networks need to address several issues concerning enterprise systems / applications integration and heterogeneous industrial information environments interoperability.

Legacy enterprise systems/applications often hinder collaboration endeavors, since, in many cases the applications are not even designed to interoperate with other applications. Another obstacle stems from the lack of standards concerning the description and orchestration of process flows across multiple systems [1].

A great number of systems and applications exist in an enterprise industrial environment covering the whole range of the enterprise system hierarchical model. This model comprises three different layers: the field / shop floor layer, the plant layer and the enterprise layer.

The field and the shop floor layers are the two lower layers of the model and comprise the applications and systems that are associated with the industrial manufacturing processes.

The enterprise layer, which comprises among others, the Enterprise Resource Planning (ERP) system, the Supply Chain Management (SCM) system, the Customer Relationship Management (CRM) system of the enterprise, utilizes information systems (ISs) applications to provide a binding of a variety of enterprise functions including human resources, inventories and financials, and at the same time a successful linking of the enterprise to its customers and suppliers.

The plant layer comprises the Manufacturing Execution System (MES) of the enterprise. MES represents ISs that address the integration of the shop floor layer and enterprise layer functionalities. Some of the MES functionalities include shop floor scheduling, production and labor reporting, and the provision of industrial process data from the shop floor layer to the enterprise layer.

The ubiquity of HTTP, TCP/IP and the Service Oriented Architecture (SOA) [2] domination has led to
the emergence of standards that allow the creation of a unifying infrastructure that makes possible the integration of different systems at whatever layer in the enterprise they reside.

Our work presented in this paper utilizes emerging standard technologies in order to achieve integration and interoperability of all the systems / applications that participate in a collaborating manufacturing process. This is accomplished by describing enterprise processes by means of workflow and using Web-Services to open up industrial systems and make their functionalities available. In addition, our approach introduces a further degree of interoperability at the system engineering level by incorporating semantic information described in ontologies related with the enterprise processes.

2. The Proposed System Architecture

Industrial communication infrastructure, comprising IP-based protocols, such as the Industrial Ethernet, the enterprise intranet and the Internet, makes possible the successful connection of the systems/applications residing in the above described enterprise layers.

However, enterprise systems/applications are characterized by considerable heterogeneity, which is attributed to a fragmentary, non-holistic view of the enterprise. This approach has made the unification of enterprise applications and systems quite difficult, let aside the successful inter-enterprise process execution. This work proposes an architecture that makes possible the successful integration of the above mentioned layers and their systems.

2.1. Methodology

Since, there is set of end-to-end processes that characterize collaborating manufacturing, our proposed methodology provides a unified solution based on a holistic view of the contemporary manufacturing enterprise, which demands the seamless integration of inter-enterprise with intra-enterprise processes. In order to satisfy this requirement, we classify processes into three categories, which also present three classes of integration.

The first category comprises processes residing in the same layer and handles layers’ systems and applications integration. Such processes usually involve more than one systems or applications of this layer that have to interoperate or exchange information.

The processes of the second category involve systems or applications that reside in different layers. These kinds of processes use information generated in different layers or initiate procedures that reside in these layers.

The third category considers processes that involve more than one different plants or even more than one different manufacturing enterprises that need to cooperate by being members of a collaborating manufacturing network.

All three classes of integration require openness of systems and applications residing on the different enterprise layer and common standardized interfaces such as web services interfaces which is utilized in our work.

2.2. Architecture

A common challenge for the system engineer is the integration of legacy and / or heterogeneous distributed systems in order to address the requirements imposed by a specific enterprise process which actually comprises a group of structured operations. A solution to this problem involves the mapping of the process tasks to the actual systems that will execute them. Depending on the way this mapping is implemented, the overall enterprise can be more or less flexible to changes.

There are three different aspects that have to be dealt by in order to make possible the successful operation of the proposed solution. These aspects comprise the efficient modeling of a process into individual tasks, the uniform exposition of applications / system functionalities following standard technologies and the intelligent association of process tasks with the exposed functionalities. Our proposed architecture addresses all three aforementioned aspects mentioned, enabling the seamless integration of enterprise systems by utilizing semantic information in order to associate in a flexible and intelligent way the enterprise processes to the actual systems. Figure 1 depicts a high level view of the proposed architecture.

2.2.1. Enterprise Process Modeling

An enterprise process has to perform a sequence of operations in order to fulfill its goals. The execution of the constituent operations is related with the necessary
data exchange between the applications and systems that reside in the different layers with reference to the aforementioned enterprise hierarchy.

A standardized tool towards the modeling of an enterprise process is workflows. Two complementary parts of a workflow are the control flow and the data flow. The control flow defines the order that the different activities in a process will be executed. The data flow specifies the flow of information between the involved activities.

The operations executed as a sequence by a workflow, stem from the existing enterprise / industrial infrastructure as well as its applications and systems. An intelligent association is thus required.

2.2.2. Enterprise / Industrial System Functionality Exposition

Standard interfaces play a crucial role regarding integration of different systems and applications residing in different enterprise layers. Web Services present a dominant XML based technological solution for standardized interfacing since they allow different systems and applications to efficiently expose their inherent functionalities, resulting in seamless and effective system integration.

Our approach utilizes Web Services as a common interface addressing vertical integration issues, meaning that different layer systems and applications are able to utilize either directly or through wrapping Web Services as the standard technology for exposing their functionalities.

2.2.3. Association of Processes and System Functionalities

The two elements mentioned above, namely workflow based process models and standard Web Service interfaces have to be intelligently associated. Statically linking workflows and Web Services suffers from a series of flaws regarding future alterations to the business process or infrastructure.

Thus, we propose the utilization of semantic information relevant to the different business processes, by means of ontologies. The advantage of utilizing ontologis is that they do not only provide semantic information but are machine readable and machine understandable as well.

The process business logic may be utilized in order to extract the necessary ontology classes. Thus, an enterprise process is described by means of both its ontology and its workflow. The decoupling of the workflow from the Web Services needed for the workflow execution through an ontological intermediate layer, results in an integrated architecture that is able to dynamically adopt to variations or changes of enterprise and industrial information systems. The actual web services that need to be called during the workflow individual step execution have to be prior associated with ontological information during the design phase. Our approach makes possible the introduction of new systems which can modify the business process without any disturbance at all. Reusability is also enhanced since process workflows are specified by ontological terms rather than system web service calls.

2.3. Implementation

The implementation of the proposed architecture involves the following functional elements:

- The Mapping Tool associating the actual web services with the enterprise process ontology.
- The Workflow Engine that is responsible for the implementation of the enterprise process business logic through a workflow.
- The Enterprise Legacy Systems that provide the needed web services.

The Mapping Tool associates the available web service arguments, found in a WSDL [3] repository, with adequate ontology class properties, and outputs the association results in an XML formatted document, which is named “activity”. Thus, an efficient mechanism is provided making the web services of an application / system easy to discover and associate with the business process semantic information contained in the ontology. Each enterprise process may be created through the composition of a number of specific “activities”. The overall composition of all “activities” associated with the workflow of a process is stored in the Workflow Engine in BPEL4WS [4] format.

The Workflow Engine is responsible for the execution of the web services described in each step of the workflow, as well as the monitoring of the progress of a workflow execution, and utilizes the BPEL4WS file created for this purpose.

For all workflow tasks the engine calls in sequence the appropriate web services using the SOAP [5] over HTTP protocol for the interconnection of the enterprise applications / systems.

3. Example Use case

In this section we apply the proposed methodology and architecture on an example use case. More specifically, we have chosen the Product Inventory Control process, being implemented by Plant Layer’s MES of a manufacturing enterprise, which interoperates with the following processes / systems as they are depicted in Figure 2:

- the Production Scheduling and Quality Assurance processes being also implemented by the MES,
- the Order Processing, Product Cost Accounting and Product Shipping Administration processes being implemented by Enterprise Layer’s ERP and
- the Warehouse Tracking System being at the Shop
Floor Layer.

We first present the process as specified by the ANSI/ISA S-95 [6] standard and then show its implementation according to our proposal.

3.1. The “Product Inventory Control Process”

Product inventory is a critical Plant’s Layer process, being implemented by the Product Inventory Control module, since its tasks include:

- keeping updated records/lists of produced end products,
- packing-out end products in accordance with delivery schedule,
- reporting on inventory to production scheduling and
- arranging physical loading/shipment of products in coordination with product shipping administration.

In addition, the operations of product inventory control module generate or modify the following information for use in other major modules, such as: finished products inventory, inventory balances, pack-out schedule, release and confirm to ship.

Its interaction with the Production scheduling, Quality Assurance, Product Shipping Administration and Warehouse Tracking modules involves respectively the following information:

- **Finished products inventory** is information that flows from the Inventory Module to Production Scheduling Module and concerns the current inventory that is available. This information can be quantity; quality and location data of inventory that is used by the Production Scheduling module in order to schedule new production or as a feedback on previously scheduled production.
- **Packet schedule** is information that flows from the Production Scheduling Module towards the Inventory Module that specifies quantities of different finished products to proceed to the packing unit to be packed and be ready for delivery to a customer.
- **Quality assurance results** is information coming to Inventory Module from the Quality Assurance Module concerning the results from QA tests performed on finished products. This information is required before the Inventory Module issues confirmation to ship.
- **Release to Ship** information flows to Inventory Module from the Product Shipping Module about the permission to ship a specific product or pack.
- **Confirm to Ship** is information expecting the Product Shipping Module from the Inventory Module prior to actually ship the product for which permission was asked for.
- **Product Movement Settings** is information flowing towards devices / systems belonging to the Shop Floor Layer and concern the physical product allocation control.
- **Settings Confirmation** is information flowing from the devices of the Shop Floor Layer to the Inventory Module verifying product allocation.

The Product Inventory Control Module comprises seven different sub-modules which utilize the above mentioned information. These sub-modules are: Inventory Supervision, Loss Control, Inventory Reporting, Product Shipping, Product Routing, Product Movement and Inventory Measurement Validation.

The **Inventory Supervision sub-module** coordinates all activities in the module, sets up transfers of material to packing unit in accordance to the pack out schedule, requests replenishment of packing materials and handles reservation and inventory update.

The **Loss Control sub-module** reports on inventory balance and losses.

The **Inventory Reporting sub-module** generates periodical reports on actual products in storage.

The **Product Shipping sub-module** sets-up and monitors product shipment to the customers in accordance with requirements from shipping module and reports confirmation of shipment.

The **Product Routing sub-module** sets-up and monitors product transfer and inventory updates.

The **Product Movement Control and the Inventory Measurement Validation sub-modules** deal with physical product allocation control and verification in the warehouses.

3.2. System Topology and required Web Services

For the purposes of our use case, we assume that the deployment of the product inventory control process
involves three departments from the Plant Layer: the Product Inventory Control Dept (PICD), the Production Scheduling Dept (PSD) and the Quality Assurance Dept (QAD), one department from the Enterprise Layer: the Product Shipping Dept. (PSAD) and one department from the Shop Floor Layer: the Warehouse Dept (WD).

Additionally, The PICD comprises one information system, the PICIS where the overall functionality of the Product Inventory Control Module is implemented. In the PICIS an Inventory Data Base is included, where all the necessary information of the finished products are stored. Respectively, the PSD comprises the PSIS, the QAD comprises the QAIS, while the PSAD comprise the PSAIS. Finally, the WD comprises the WaTS.

In order to make possible the implementation of our proposed architecture, appropriate web services are needed to expose the functionalities of the above described information systems. More specifically, PICIS supports the following web services methods: get_finished_products(), send_packet_schedule(), get_finished_product_for_quality_check(), send_quality_check_results(), get_confirmation_for_shipping(), send_release_for_shipping(), send_pr_movement_setup(), get_pr_movement_confirmation(). PSIS supports the send_finished_products() and get_packet_schedule() methods, QAD supports the send_finished_product_for_quality_check() and the get_quality_check_results() methods, PSAD supports the send_confirmation_for_shipping() and the get_release_for_shipping() methods and finally, WaTS supports get_pr_movement_setup() and send_pr_movement_confirmation() methods.

3.3. Ontology

The Product Inventory Control process, as defined in section IVA, comprises the terms needed in order to generate the ontology for the associated business logic. The ontology that will be generated is according to the OWL standard [7], using the Protégé 2.1.2 tool [8]. The ontology, which we call ProductInventoryControlOnto, consists of seven classes. The classes “Customer” and “Product” refer to common modules and roles of the enterprise while the classes “FinishedProducts”, “PackingInformation”, “ProductMovementSetup”, “ProductShipping” and “QualityCheck” describe concepts and entities used exclusively during the process execution. The ontology schema accompanied with class properties and their relationships is shown in Figure 3.

3.4. Mapping Web Services to the Ontology

The first step, during the design-configuration phase, is the mapping of web services to relevant ontology terms. This mapping involves the association of web service method request and response arguments to object or data type properties of an ontology class. We
call these sets of associations, activities. For each activity one or more selected web services methods are selected and their arguments are associated with the appropriate ontology terms.

As an example, we provide the mapping for the interrelated pair of methods: send_pr_movement_setup() and get_pr_movement_setup that are exposed by the PICIS and WaTS respectively, as shown in Table 1. We call this activity “Product Movement Settings”.

According to our methodology a list of activities for the enterprise systems should be generated. Each activity is given a name for distinction purposes. Below we present the created activities for the web services methods presented in IV B:
1. “Finished Products Inventory”
2. “Packet Schedule”
3. “Require Quality Assurance”
4. “Quality Assurance Results”
5. “Release To Ship”
6. “Confirm To Ship”
7. “Product Movement Settings”
8. “Settings Confirmation”

The Mapping Tool provides, for each activity, an XML/RDF output file that includes the OWL ontology and the associated mappings in a custom namespace. In addition, each mapping in the activity is characterized by the attribute called “scope”. The possible values that the “scope” attribute can take are: “input”, “output” and “internal”. The value “input” characterizes the mappings that represent input data to the activity, the value “output” characterizes the mappings that represent output data of the activity and the value “internal” characterizes the mappings that represent data used internally in the activity, which pass automatically information from one method to another.

Figure 4 below depicts the resulting XML/RDF output file for the “Product Movement Settings” activity.

3.5. Workflow design

During the last step of the design phase in our proposed methodology, the process workflow should be developed. This workflow, as depicted in Figure 5, comprises the activities series that should be executed during the deployment of the Product Inventory Control process.

4. Conclusions

The effect of web-enabled technologies is to greatly increase the channels of communication within an enterprise. The manufacturing model is no longer confined to limited and rigorous communication channels. Information from the lower shop floor layer
can be readily linked to plant floor and enterprise layers and to collaborative manufacturing networks. Web technologies tend to flatten the hierarchical structure of traditional manufacturing enterprise information systems models.

The presented work makes use of these advancements and proposes an architecture that combines Web services, workflows and ontologies in order to efficiently integrate diverse systems and applications independently of the layer in a collaborative environment they may reside in.

References
