



**HAL**  
open science

# Ontology for Service-Based Control of Production Systems

Elisa Negri, Luca Fumagalli, Marco Macchi, Marco Garetti

► **To cite this version:**

Elisa Negri, Luca Fumagalli, Marco Macchi, Marco Garetti. Ontology for Service-Based Control of Production Systems. IFIP International Conference on Advances in Production Management Systems (APMS), Sep 2015, Tokyo, Japan. pp.484-492, 10.1007/978-3-319-22759-7\_56 . hal-01431136

**HAL Id: hal-01431136**

**<https://inria.hal.science/hal-01431136>**

Submitted on 10 Jan 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Ontology for Service-Based Control of Production Systems

Elisa Negri<sup>1</sup>, Luca Fumagalli<sup>1</sup>, Marco Macchi<sup>1</sup> and Marco Garetti<sup>1</sup>

<sup>1</sup> Department of Management, Economics and Industrial Engineering  
Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano.  
{elisa.negri, luca1.fumagalli, marco.macchi, marco.garetti}@polimi.it

**Abstract.** The paper illustrates a production systems ontology that models the discrete manufacturing, process production and the logistics domains. This ontology is used to allow semantic interoperability within a control architecture based on semantically-enriched Web Services that has been developed within the European funded project eScop. This architecture would facilitate the responsiveness and agility of the manufacturing companies, helping them to be more competitive thanks to the higher flexibility and re-configurability of their production systems.

**Keywords:** Manufacturing Systems Ontology · Interoperability · Open Automation · Semantic Representation · Flexibility · Re-configurability.

## 1 Introduction

Researchers agree on the importance of the benefits that ontologies, being conceptual representations of different domains [1], provide to the fields when they are implemented. One of the fields in which ontologies seem promising, but are not yet widely applied, is the industrial engineering. Ontologies may provide a representation of the production systems as an aid for industrial engineering uses. Indeed, it is a long time since researchers have expressed the need for an ontology to contribute to this field: already in the nineties, Schlenoff stated the main benefits that come from an application of such an ontology [2]. The ontologies are developed for different uses, in fact each is characterized by a different representation potential and level. There are some that are high level, or “foundational”, that express generic concepts that should be shared among different fields and that are used to bridge among different ontologies; their importance is remarked by [3]. An example of these is the DOLCE upper ontology [4]. Others are more specific for a domain and a use, as shown in a recent work by Fortineau et al.: in this work, it is made clear how wide is the range of applications of ontologies within the industrial engineering: they cover product, services and processes and their life cycles (from development phases to end of use) [5]. In the mentioned work, there is a strong prevalence of ontologies for the product development, because activities involved in this phase require the collaboration of people from different fields and with different expertise and backgrounds: in such a context, an enormous benefit brought by ontologies is the creation of a shared, explicit and common vocabulary to express the required concepts [6, 7]. The potential of ontologies in the industrial engineering field is not limited to this role of vocabulary sharing instrument, in fact they can add semantics to the information that is exchanged between different systems and applications at different levels [8, 9], this is the basis for interoperability, defined by Ide and Pustejovsky as when “two systems have the ability to automatically interpret exchanged information meaningfully and accurately in or-

der to produce useful results via deference to a common information exchange reference model”, in other words, “the content of the information exchange requests are unambiguously defined: what is sent is the same as what is understood” [10]. The application proposed in this work takes this perspective, by exploiting the semantic content of ontologies for interoperability into service-based communication networks of embedded systems. A recent ontology of the production systems is presented to this end. This comes as an evolution of early works carried out at Politecnico di Milano in the direction of the development of a conceptual representation of the production systems [11, 12], basing also on other works [13, 14, 15]. These can be in fact considered as precursors of more recent production systems ontologies, such as the P-PSO (Politecnico di Milano – Production Systems Ontology), an object-oriented structured representation of the domain of the manufacturing systems described in [16, 17]. P-PSO is aimed to be a meta-model of the manufacturing systems domain, since it specifies the entities (objects) a manufacturing system is composed of, their attributes (parameters) and the relationships between them, and the necessary constraints, thus defining a standard way to describe any manufacturing system. P-PSO is developed and evolved to be “a common information exchange reference model” for a control architecture based on semantically-enriched Web Services, developed within the European funded project eScop.

This paper presents the evolution of the P-PSO (section 2) and its deployment within a service-based shop floor control architecture (section 3); eventually (section 4), it frames these results within the wider context of higher flexibility and re-configurability in production systems, thus motivating its use in relationship to the industrial engineering field.

## 2 MSO Ontology Illustration

The P-PSO ontology, mentioned in the previous section, has evolved into the Manufacturing System Ontology, in short MSO, that is a domain ontology for the representation of production systems, that covers the domains of logistics, discrete and process production. MSO model includes many concepts from the P-PSO, at the same time presenting differences on three levels:

(i) on the domain level: the MSO includes the domains of process industry and logistics that were not covered in the P-PSO (that only represents discrete manufacturing systems); this can be achieved thanks to the fact that the high level domain-crossing classes are defined for all types of industry, while the specific industry classes are defined as specialization of the high level common objects;

(ii) on the use level: as it is explained in section 4, the MSO is used for the control of the production system; while the P-PSO is a general taxonomy not built for a specific use, but it can support design, simulation, planning and scheduling, performance assessment and data integration in the manufacturing field [17];

(iii) on the modelling level of the control and visualization aspects:

- in P-PSO the control aspect defines all the entities and relationships that exist among the concepts that are needed to perform control, in the MSO instead the control is only defined at a conceptual level, because it is just a knowledge storage support for a control software that acts outside of the ontology and only retrieves

information from it: therefore, there is no need to represent how the control software works, also because this is not standardized enough to have a generalized data structure;

- P-PSO does not have any concept related to the visualization and display of information; while in MSO this is present, providing modelling of elements that are needed for a proper management of visualization of the manufacturing systems to enhance awareness of the system by operators.

As in P-PSO, the developed ontology MSO covers different aspects within the manufacturing and process industry domain: the physical aspect, the technological aspect, the control aspect, the visualization aspect, that will be later detailed. The classes of the MSO ontology are represented in Figure 1, as a screenshot from the used ontology editor, which shows them in alphabetical order. Therefore the different aspects do not have a clear visual separation in this view, but can be understood from the names of the classes. As mentioned above, each of these classes has attributes, relationships and constraints that are necessary to define the overall structure of the manufacturing systems domain but are not shown in Figure 1 due to space constraints.

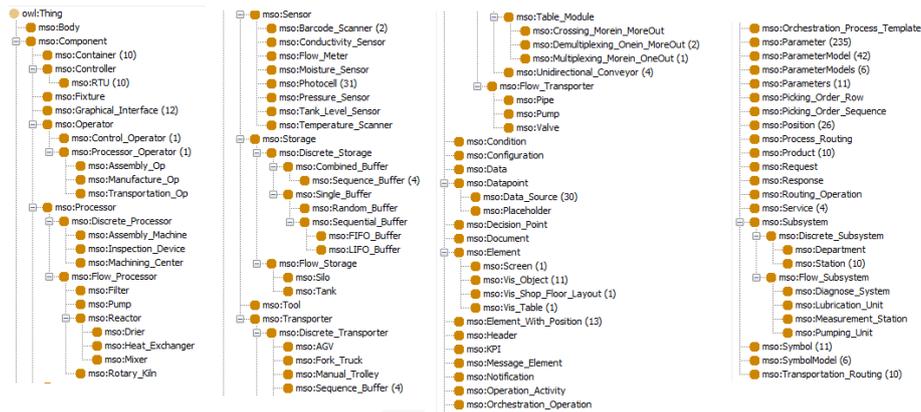


Figure 1. The MSO ontology classes

## 2.1 The Physical Aspect

The basic objects of the physical aspect of the MSO ontology are the “*Component*” class (that represents a physical object in the production system) and the “*Subsystem*” class (composed of more than one components together).

The *Subsystem* class is specialized into sub-classes that are typical of the process production or of the discrete manufacturing. An example of discrete subsystem is the Station, the place where the processor or an operator is physically working.

The *Component* class can be specialized into sub-classes: (i) *Transporter*, any type of conveyor and movement device performing a transportation (i.e. logistics) function, moving material along places of the manufacturing process; examples are: AGVs, conveyors, fork trucks; (ii) *Processor*, identifies entities performing a manufacturing or process function by using an energy source and/or an operator activity; examples are: machining centers, inspection devices, assembly machines; (iii) *Stor-*

*age*, representing any entities performing a storage (i.e. logistics) function, it means that they are able to keep material for later use in the manufacturing process; types of storage are: tanks, buffers, automated storage and retrieval systems; (iii) *Operator*, any type of person that perform activities in the manufacturing system and interact with other components; this class needs a specific approach, since operators can carry out different types of operating activity (for instance processing, transport, inspection, maintenance) and also carry out no-less-important control activities, like support and supervision over any type of component; (iv) *Sensor*, any type of sensing device used to capture the current status of a physical variable; (v) *RTU*, Remote Terminal Unit, any type of electronic control device that interfaces physical objects of the controlled production system; (vi) *Tool*, entities used by a processor to perform an operation on a product; (vii) *Fixture*, entities that are used to position, hold, support, locate and clamp the product in a three dimensional space; (viii) *Container*, representing the box in which products lie when they flow into the system. The *Transporter*, *Processor and Storage* classes, through the specialization association, are sub-divided into flow type and into discrete type, in order to address both the process industry and the discrete manufacturing industry, respectively.

## **2.2 The Technological Aspect**

In the technological aspect, the developed ontology separates two different routings concepts: the *Transportation routing* and the *Process routing*. The former represents the physical displacement cycle that the product or the material must perform in order to finish its production: the Transportation routing is therefore composed of *Stations* (subsystems that were defined in the previous section 2.1) that the product or material must visit in order to be completed. The Process routing instead defines is the production cycle in a more traditional sense: it is composed of the operations to be performed and defines the order in which they must act on the product or material. There is a link between the two routings, because specific operations are performed in specific stations. However, while an operation is linked to only one station, there can be more than one operation in each station, for this reason it is important to keep the two routings conceptually separated. The ontology also represents the fact that each production operation is composed of one or more elementary activities, featured with a number of information and characteristics that must be stored (such as needed instructions, needed equipment, required operator skills, input materials and others) and, in order to know if one activity has finished and the next one start, a condition must be met (which is modeled and stored in the ontology).

## **2.3 The Control Aspect**

The control aspect includes all the classes that store the knowledge and the information required for the proper activities of the control software. For instance, it stores information about the product, the customer orders (due dates, quantities, etc.) and about the services that must be called to perform an operation and service parameters that describe the services, such as service URL and type, the meaning of the service and its availability. The role of services in the control of the production system using this ontology will be explained in section 3.

## 2.4 The Visualization Aspect

The visualization aspect supports the human-machine interface (HMI) screens by presenting the information about the system in a clear and effective manner using graphical representation and retrieving the necessary information from the ontology. The ontology stores all the information needed for the visualization of the system and it can be retrieved or updated by the visualization provider based on queries from visualization agent. The visualization knowledge in the ontology includes the information on composition of the screen and the link between the visual elements and real components or data of the system. All the *Components* in the system have a graphical interface which is composed of *Elements* that represent them in the visualization screen. The *Element* class represents the visual elements like screens for operator display, shop floor layout, table and so on. The elements are linked to some position on the screen, that are defined in terms of X-Y coordinates, or in terms of rows and columns. Moreover, elements have visualization parameters like sensor values, temperature, etc. that are needed to show information interesting to the operators.

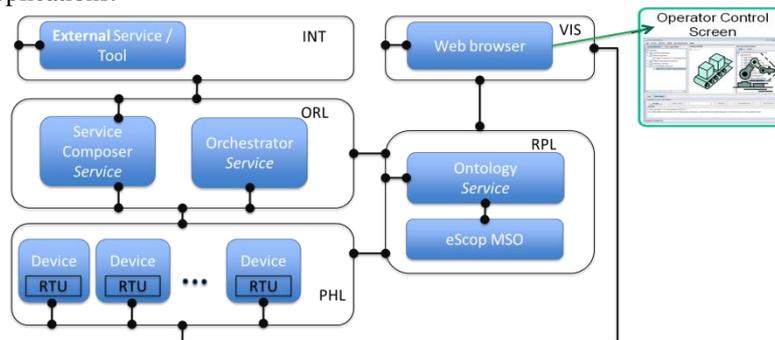
## 3. Role of MSO in the eScop Architecture

The developed ontology, presented in section 2, has a practical implementation within the scope of the European Artemis funded project “eScop”, acronym for Embedded systems for Service-based Control of Open manufacturing and Process automation. This project is aimed at proposing a solution for the current problems of production system flexibility and even agility, by allowing a more rapid reconfiguration and an easier integration of production system elements at shop-floor control level. This is done with the inclusion of semantic content into the control level of the devices and applications through the use of the developed ontology presented in section 2, that allows interoperability among devices from different vendors.

The core idea of the project is to put embedded systems in communication with open standards, such those of the web, in a Service-Oriented Architecture (SOA) network where each device offers Web Services on the network. This allows to reach the open automated manufacturing environment that has been depicted by [18].

The role of the developed ontology in this architecture is that the SOA control architecture can be automatically configured thanks to the continuously updated ontological content, while the embedded systems allow this architecture to operate in the production system environment. Figure 2 presents the software architecture of the control system employed in the eScop project. From the picture, it can be understood that the layers needed for the control are five. *Physical Layer (PHL)*, composed of the physical devices and their service-enabled device control units that put them in communication with the rest of the shop floor and control the equipment at local level. *Representation Layer (RPL)*, composed of the developed MSO Ontology model in which the knowledge about the configuration and the current status of the production system is stored, and of the Ontology Service that is the software application that allows to expose the ontological knowledge as a service in the network, in particular it is the Ontology Service that queries the ontology to retrieve the information within stored. *Orchestration Layer (ORL)*, incorporating the role of the control software that invokes, orchestrates and coordinates the services offered by the devices in the system

to perform the necessary production activities; it is made of 2 components: i) Service Composer, which receives task needs and decides how to orchestrate the system in order to fulfill task needs, and ii) Orchestrator service which does the orchestration process to ensure its successful execution. *Visualization Layer (VIS)*, used for the human-machine interface with the operator on the line. *Interface Layer (INT)*, completing the architecture, being the module that communicates with external software and applications.



**Figure 2.** Positioning of the ontology into eScop architecture

In such an architecture the role of the developed ontology is clear: it receives information from the field devices in the physical layer and supports the orchestration layer with knowledge that is necessary to make the proper control decisions; in addition it supports visualization by providing the necessary information to be displayed for the operators (please see the connectors that link one layer with the others in the figure 2).

#### 4. Conclusions

The paper proposes a new ontology for the production systems domain and a possible application in the context of a service-based control architecture of the production systems. The result of the implementation of the Manufacturing Systems Ontology presented in sections 2 and 3 is a modular, fully open software solution for the operational control of manufacturing equipment in production systems allowing the many benefits of remarkable industrial relevance: (i) easier and faster commissioning of new plants, (ii) support of the “plug & produce” inclusion of new equipment, (iii) replacement of traditional control based on hierarchical hardware architectures by a single level cohort of embedded systems and a series of software control levels. These are in line with the progressing research needs to reach higher responsiveness and agility in manufacturing companies: companies are searching ways to be more flexible and even more agile to reconfigure their production systems in a quicker way, in order to face the increasing competition at a global scale and to face the more demanding consumers requests in a timely fashion [19, 20]. The automation paradigm may be one of the facilitating factors for higher flexibility and re-configurability because an automated control software update, based on the semantic content of the ontology, can be achieved that does not require the intervention of the human programmer whenever the physical production system undergoes any change [21]. For

this reason, research in this direction has enormous industrial interests and many possibilities for improvement can be considered. For example, it is worth envisioning even the use of the MSO in the context of a scalable production, released by means of small production units integrating their own logistics, which is a concept drawn within Factory of the Future visions. Another possible application could be to enlarge the potential of MSO towards enhancing not only the production system flexibility – which is a base target – but also the agility and, thus, re-configurability of an organization of production plants operating in changing environments and spread in a given geographical scale, defined according to the target market.

Future steps include the application of the MSO ontology and the open automation architecture for the control of production systems to industrial contexts in the discrete manufacturing, in the process production and in the logistics to validate the expected benefits discussed in this paper. Further work can be directed at studying how to enlarge the distributed systems basis on which the developed ontology can be applied.

**Acknowledgements.** The research leading to these results has received funding from the ARTEMIS Joint Undertaking under grant agreement n° 332946 and from the Italian Ministry of Education, Universities and Research (MIUR), correspondent to the project shortly entitled *eScop, Embedded systems for Service-based control of Open Manufacturing and Process Automation*.

## 5. References

1. T. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," *Int. J. Hum. Comput. Stud.*, vol. 43, pp. 907–928, 1995.
2. C. Schlenoff, R. Ivester, D. Libes, P. Denno, and S. Szykman, *An Analysis of Existing Ontological Systems for Applications in Manufacturing and Healthcare*. 1999.
3. S. Borgo and P. Leitão, "The role of foundational ontologies in manufacturing domain applications," *Move to Meaningful Internet Syst. 2004 CoopIS, DOA, ODBASE*, pp. 670–688, 2004.
4. A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider, "Sweetening ontologies with DOLCE," *Knowl. Eng. Knowl. Manag. Ontol. Semant. Web, Lect. Notes Comput. Sci.* vol. 2473, pp. 223–233, 2002.
5. V. Fortineau, T. Paviot, and S. Lamouri, "Improving the interoperability of industrial information systems with description logic-based models—The state of the art," *Comput. Ind.*, vol. 64, no. 4, pp. 363–375, May 2013.
6. K.-Y. Kim, D. G. Manley, and H. Yang, "Ontology-based assembly design and information sharing for collaborative product development," *Comput. Des.*, vol. 38, no. 12, pp. 1233–1250, Dec. 2006.
7. C. E. Catalano, E. Camossi, R. Ferrandes, V. Cheutet, and N. Sevilimis, "A product design ontology for enhancing shape processing in design workflows," *J. Intell. Manuf.*, vol. 20, no. 5, pp. 553–567, Jul. 2008.
8. A. K. Y. Wong, P. Ray, N. Parameswaran, and J. Strassner, "Ontology mapping for the interoperability problem in network management," *Sel. Areas Commun. IEEE J.*, vol. 23, no. 10, pp. 2058–2068, 2005.
9. H. Panetto and A. Molina, "Enterprise integration and interoperability in manufacturing systems: Trends and issues," *Comput. Ind.*, vol. 59, pp. 641–646, 2008.
10. N. Ide and J. Pustejovsky, "What Does Interoperability Mean , Anyway? Toward an Operational Definition of Interoperability for Language Technology," *Proc. Second Int. Conf. Glob. Interoperability Lang. Resour.*, 2010.

11. A. Bartolotta, E. Corradi, and M. Garetti, "Developing an ontology for the modelling of manufacturing systems," in IFIP International Enterprise Modeling Conference, 1999.
12. A. Bartolotta and M. Garetti, "Object-oriented representation of manufacturing systems: State of the art and perspectives," in *Advances in Production Management Systems IFIP — The International Federation for Information Processing*, 1998, pp. 195–205.
13. M. Colledani, W. Terkaj, T. Tolio, and M. Tomasella, "Development of a Conceptual Reference Framework to Manage Manufacturing Knowledge Related to Products, Processes and Production Systems," in *Methods and Tools for Effective Knowledge Life-Cycle-Management*, A. Bernard and S. Tichkiewitch, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 259–284.
14. M. Cai, W. Y. Zhang, and K. Zhang, "ManuHub: A Semantic Web System for Ontology-Based Service Management in Distributed Manufacturing Environments," *IEEE Trans. Syst. Man, Cybern. - Part A Syst. Humans*, vol. 41, no. 3, pp. 574–582, May 2011.
15. S. Lemaignan and A. Siadat, "MASON: A proposal for an ontology of manufacturing domain," *Proc. IEEE Work. Distrib. Intell. Syst. Collect. Intell. Its Appl.*, pp. 6–11, 2006.
16. M. Garetti and L. Fumagalli, "Role of ontologies in open automation of manufacturing systems," in *Proceedings of the XVII Summer School in Industrial Mechanical Plants - 12/9/2012-14/9/2012*, Venice, Italy, 2012.
17. M. Garetti and L. Fumagalli, "P-PSO Ontology for Manufacturing Systems," *Inf. Control Probl. Manuf.*, vol. 14, no. 1, pp. 449–456, 2012.
18. M. Garetti, L. Fumagalli, a. Lobov, and J. L. Martinez Lastra, "Open automation of manufacturing systems through integration of ontology and web services," *IFAC Proc. Vol.*, pp. 198–203, 2013.
19. J. Meredith and U. Akinc, "Characterizing and structuring a new make-to-forecast production strategy," *J. Oper. Manag.*, vol. 25, pp. 623–642, 2007.
20. F. Salvador and C. Forza, "Configuring products to address the customization-responsiveness squeeze: A survey of management issues and opportunities," *Int. J. Prod. Econ.*, vol. 91, pp. 273–291, 2004.
21. I. Delamer and J. Lastra, "Ontology modeling of assembly processes and systems using semantic web services," in *Industrial Informatics, 2006 IEEE International Conference on*, 2006, pp. 611–617.