Computers in Industry 88 (2017) 19–34

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

Modelling internal logistics systems through ontologies

Elisa Negri*, Sara Perotti, Luca Fumagalli, Gino Marchet, Marco Garetti[1]

Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133 Milano, Italy

ARTICLE INFO

Article history:
Received 27 June 2016
Received in revised form 30 January 2017
Accepted 10 March 2017
Available online 27 March 2017

Keywords:
Internal logistics
Warehousing
Ontology
Semantic representation
Domain modelling

ABSTRACT

Industry is facing an era characterised by unpredictable market changes and by a turbulent competitive environment. The key to compete in such a context is to achieve high degrees of responsiveness by means of high flexibility and rapid reconfiguration capabilities. The deployment of modular solutions seems to be part of the answer to face these challenges. Semantic modelling and ontologies may represent the needed knowledge representation to support flexibility and modularity of production systems, when designing a new system or when reconfiguring an existing one. Although numerous ontologies for production systems have been developed in the past years, they mainly focus on discrete manufacturing, while logistics aspects, such as those related to internal logistics and warehousing, have not received the same attention. The paper aims at offering a representation of logistics aspects, reflecting what has become a de-facto standard terminology in industry and among researchers in the field. Such representation is to be used as an extension to the already-existing production systems ontologies that are more focused on manufacturing processes. The paper presents the structure of the hierarchical relations within the examined internal logistics elements, namely Storage and Transporters, structuring them in a series of classes and sub-classes, suggesting also the relationships and the attributes to be considered to complete the modelling. Finally, the paper proposes an industrial example with a miniload system to show how such a modelling of internal logistics elements could be instanced in the real world.
© 2017 Elsevier B.V. All rights reserved.

1. Introduction

Recently companies are facing frequent and unpredictable market changes driven by global competition. A key challenge in such a turbulent environment is for companies to be responsive, i.e. companies must be able to react to all kind of changes rapidly and cost-effectively if they want to remain competitive [1]. In order to achieve responsiveness, manufacturing companies are using novel technologies and redesigning their existing production systems to meet the new requirements: in particular, the rapid configuration of new systems and rapid re-configuration of the existing ones are among the most promising challenges to reach a high level of responsiveness [1].

The rapid (re)configuration of production systems has been studied since many years and great advancements have been achieved in terms of enabling technologies and approaches, that offered a high level of automation and flexibility. A keystone approach for flexibility and rapid (re)configuration is surely the modularity of the production system itself [2], that perceives the system as a composition of single modules. These are recognised as elements that have a specific function and could be controlled autonomously. The modules offer a high level of flexibility, at physical and mechatronic levels with standard interfaces, however the control does not present an equal level of modularity: it is still thought as a whole software system that manages the entire system in all its parts. Different studies [3–5] have suggested that ontologies could represent the needed knowledge base to support flexibility and modularity of production systems, when designing a new system or when reconfiguring an existing one (e.g. to easily update the control software system to be compliant with the specific production system [6], or to share a vocabulary among developers and designers).

The production field, and in particular reference logistics-related aspects, have been studied since many years, so that literature and applications have matured to a great extent offering de-facto standards in the industrial solutions with practical alignment in terms and concepts among academics and practitioners. In fact, numerous research works have proposed taxonomies, classifications and definitions in the field of logistics (e.g. [7–11]). However, their main aim was to classify the available

* Corresponding author.
E-mail addresses: elisa.negri@polimi.it (E. Negri), sara.perotti@polimi.it (S. Perotti), luca1.fumagalli@polimi.it (L. Fumagalli), gino.marchet@polimi.it (G. Marchet), marco.garetti@polimi.it (M. Garetti).
[1] Deceased.

http://dx.doi.org/10.1016/j.compind.2017.03.004
0166-3615/© 2017 Elsevier B.V. All rights reserved.

solutions (e.g. for material handling, storage, and picking) rather than to address modularity and (re)configurability issues.

The contribution that this paper would like to offer is an ontology of the internal logistics (i.e. warehousing) elements, to be used as semantic support to the modularity of production systems. The objective of the paper is to model internal logistics systems in an ontology that could be used as knowledge base for different applications towards system modularity (to name one, the semantics at the basis of a control systems that flexibly builds on the aggregation of single modules). To this purpose, the authors propose both a taxonomy of the internal logistics elements and a way to model them into an ontology.

To sum up, the drivers that push this research work are:

- The need for a shared and structured definition and modelling of the internal logistics terminology, that reflects the concepts and taxonomies already existing in the scientific community and in the industry;
- The need to build a model to support logistics systems modularity, that is going to increase even more according to the mentioned technological trends;
- The need to develop a model for the internal logics of the logistics system structure that is machine processable and that allows the integration of knowledge within automated systems.

The remainder of the paper will present a literature review and background in Section 2; the research objectives will be explicated in Section 3; the research method is described in Section 4; Section 5 and its subsections present the proposed model; Section 6 is an application example of how to use the ontological model to describe an industrial system. Conclusions are reported in Section 7.

Table 1
Research on ontologies for logistics: examples.

Ontology subject	Reference	Paper	Year	Use of ontology	Main focus		Perspective	
					Supply Chain	Internal Logistics/ Warehousing	Process	Physical resources
Logistics only as minor aspect	[49]	Lemaignan et al.	2006	Distributed manufacturing control (and automated cost estimation)		X	X	x, poorly described
	[2]	Alsafi and Vyatkin	2010	Distributed manufacturing control		X	X	
	[39]	Kiritsis	2011	Product data technology for Product Lifecycle Management (PLM)	X		X	
	[50]	Nadoveza and Kiritsis	2014	Information and data management	X	X	X	
Logistics as main body of the ontology	[56]	Matheus et al.	2005 (1)	Situational awareness (i.e. ontology to represent the relevant rules and relationships to be monitored)	X		X	x, poorly described
	[57]	Matheus et al.	2005 (2)	Situational awareness (i.e. ontology to represent the relevant rules and relationships to be monitored)	X		X	
	[58]	Fayez et al.	2005	Support to supply chain simulation	X		X	x, poorly described
	[59]	Himoff et al.	2006	Information support to logistics scheduler	X		X	X
	[60]	Lian et al.	2007	Information support to event notification system		X	X	
	[61]	Chandra and Tumanyan	2007	Decision support for scheduling in multi-state steel manufacturing processes	X		X	
	[62]	Gonnet et al.	2007	Sharing a precise meaning of the information exchanged during communication among the many stakeholders involved in the Supply Chain	X		X	x, poorly described
	[48]	Leukel and Kirn	2008	Automated data integration along the supply chain	X		X	
	[63]	Ha et al.	2008	Support to Decision Support Systems (DSS)		X	X	
	[64]	Gimenez et al.	2008	Inter-enterprise integration of logistics information flows	X			
	[65]	Ye et al.	2008	Semantic integration among heterogeneous supply chain information systems	X		X	
	[66]	Park et al.	2008	Inter-enterprise integration of logistics information flows	X		X	
	[67]	Tsou	2008	Knowledge sharing and communication	X			X
	[68]	Nie et al.	2009	Mapping of different logistics ontologies for e-commerce	X		n.a.	n.a.
	[3]	Hoxha et al.	2010	Decentralised control and planning of logistics systems	X		X	x, poorly described
	[69]	Zhang and Tian	2010	Logistics information continuity in information platforms	X		X	
	[70]	Bonfatti et al.	2010	Automated logistics document integration among legacy systems	X		X	
	[54]	Grubic and Fan	2010	Inter-enterprise integration of logistics information flows	X		n.a.	n.a.
	[71]	Chi	2010	Tracing of suppliers and inbound freight	X			
	[72]	Grubic et al.	2011	Modelling and quantitative analysis of supply chain processes	X		X	
	[53]	Sakka et al.	2011	Transforming Supply Chain Operations Reference (SCOR) model	X		X	
	[4]	Scheuermann and Hoxha	2012	Support to flexible IT architectures	X		X	x, poorly described
	[73]	Lu et al.	2013	Integrating product information within the entire supply chain	X		X	
	[52]	Yang et al.	2013	Support to Decision Support Systems (DSS)	X		X	
	[23]	Scheuermann and Leukel	2014	Diverse range of applications: support to supply chain modelling, planning, scheduling, simulation, and information integration along the supply chain	X		n.a.	n.a.
	[51]	Gonzalez-Rodriguez et al.	2015	Inter-enterprise integration of logistics information flows	X		X	

2. Theoretical background on ontologies and modelling in manufacturing and logistics

In the Information Technology (IT) field, the ontology concept describes an “explicit specification of a conceptualisation that facilitates knowledge sharing and reuse”, where a conceptualisation is an abstract, simplified view of the world that we wish to represent for some purposes [12]. Ontologies are the elements that can capture the intrinsic conceptual structure of a domain and create a vocabulary to represent it [13] with an object-oriented description that expresses taxonomies and semantically rich relationships among concepts, in this way also allowing to include knowledge into software and automatic systems. Ontologies are knowledge bases with peculiar characteristics with respect to the traditional databases technologies, among which the knowledge content and the knowledge structure are expressed in the same way, and, most of all, they allow interoperability and reasoning capabilities, i.e. they may act as the “vocabulary” for the communication among different software systems and allow deduction of new knowledge [14–18]. The potential benefits of the inclusion of semantics in any application field through ontologies are great. By their nature, ontologies do not have a specific application domain, but they may be the means to represent knowledge of any domain, in order to make it shared, explicit and formal [12].

In the production systems domain, solutions under development in the last years present extremely high levels of automation and complexity, also incorporating internal logistics systems. For this reason, a shared way to represent the production systems is now necessary that could become a standard description of the objects constituting them.

Much work on ontologies has been already performed with a specific focus on manufacturing aspects in production. In fact, already in 1999, it has been recognised that applications of ontologies in manufacturing are justified for three main reasons: unambiguous communication, shared terminology and semantic alignment and ad-hoc access to single objects in distributed industrial information infrastructure [19]. The attempts to describe the production elements of the manufacturing domain into ontologies span from the control, to the design, simulation, production planning and life cycle data integration of the production systems [20–24].

In literature, the uses of ontologies in the manufacturing domain can be classified as:

- Support to rapid re-configuration of manufacturing systems (as examples: [2,25–27]);
- Integrated modelling of manufacturing systems (as examples: [21,28–30]);
- Inter- and intra-enterprise interoperability for different systems (as examples: [31–38]);
- Knowledge sharing and reuse (as examples: [39–46]);
- Reasoning capabilities to deduct new knowledge (as example: [47]).

However, these have still not met a wide recognition as standardised representation of the production elements within the manufacturing domain for the production elements. Moreover, the need for an even higher modularity level has made it necessary to represent in the shared ontological model not only strictly manufacturing elements but also aspects of internal logistics (i.e. warehousing) that have not been considered enough in previous works [48]. This is needed to have a fully flexible and modular systems.

As anticipated, the ontological modelling of logistics aspects has been less investigated by research communities [4,48]. To offer a clear understanding of previous research on these topics, Table 1 presents some references related to existing ontologies in the logistics arena, listed in chronological order. Specifically, in the table papers are classified accordingly to the following criteria:

- ontology main subject, i.e. manufacturing – with logistics as a sub-topic – or logistics (first group and second group in the table, respectively);
- ontology main focus, i.e. supply chain versus internal logistics/ warehousing;
- ontology perspective, i.e. process versus physical resources.

Also, the objective of using ontologies is reported for each examined work. Looking at this latter aspect, interestingly the majority of papers propose ontologies for information and data integration among different systems or supply chain partners (i.e. they exploit the interoperability potential of ontologies), however some studies envisioned the need for a more wide-ranging logistics ontology to achieve higher flexibility and therefore be competitive in today’s global business environment, that is in line with what the authors of the present paper agree upon.

As an example, [3] proposes an ontology for decentralised Information Technology (IT) solutions for planning and control to have greater flexibility and re-configurability of supply chain. In this paper, the ontology is key to overcome the increasing structural complexity and dynamicity of the modern logistics systems, which arises from trends in the global economy, supply chains decomposition and individual customer demands. The contribution is a formal semantic knowledge model of the information in the logistics domain using ontologies to enable automated and intelligent techniques for discovery, ranking, execution and efficient composition of services into more complex and flexible logistics processes.

Also, [4] recognises the vital role of IT and especially the need for flexible IT architectures and intelligent approaches to flexibly provide logistics capabilities in order to gain competitive advantage. The authors also envision the use of Semantic Web Technologies (SWT) and Service-oriented Computing (SOC) to enhance logistics processes, and in particular they focus on Supply Chain Management (SCM), being SOC a paradigm for distributed, potentially cross-organisational software systems, and aims at rapidly and easily providing applications by combining single services to enable flexible business processes.

Overall, as also pointed out by [4], in some works production ontologies present both manufacturing and logistics aspects, but the latter are often considered as a sub-topic, mentioned within a more complete representation of the manufacture (e.g. [2,39,49,50], to name few of them) [4]. However, some research contributions have been also found where the ontology modelling is adopted specifically for logistics (e.g. [23,51–53]). In these studies, the term ‘logistics’ can either refer to ‘external’ logistics, i.e. connected to the flow of goods and related-information along the supply chain, or to ‘internal’ logistics, i.e. warehousing issues, such as material handling, storage and picking within a warehouse or a plant. In general, the supply chain perspective has been more often adopted than the internal logistics one. In fact, in many of the examined papers ontologies have been either considered to achieve higher data integration among companies operating in the same supply chain, or for ensuring data continuity along the life cycle of a certain component. This is in line with the current international scenario of global supply chain-based competition, thus making information exchange among supply chain partners a competitive tool [54]. As far as internal logistics is concerned, only very few contributions are explicitly addressing ontologies for warehousing applications, as reported in Table 1. It should be noted that, to a different extent, a number of taxonomies were also found in logistics-oriented international peer-reviewed journals, addressing material handling, storage, and picking systems (e.g., [7,10,11,55]). However, their main aim was to provide a classification of the existing internal logistics solutions rather than offering a structured representation of their components and related features, as offered by ontologies. As such, these taxonomies were excluded from Table 1.

Finally, another interesting consideration could deal with the scope of the proposed ontological representations of supply chains or internal logistics systems. These systems can be described from a physical point of view, i.e. as a composition of various logistics resources (such as transporters, storage modules, vehicles), or from a process perspective (i.e. describing the interactions needed to fulfil a transportation or logistic activity). The majority of the examined ontological models adopt the second perspective, considering logistics operations and order management-related modelling, but often neglecting the logistic resources that carry out these operations.

Based on the analysis of the current literature on the topic, it is clear that the few existing logistics ontologies are mainly addressing inter-organisational logistics for supply chain applications. They mainly focus on process aspects, often providing abstract concepts and a very small number of concepts and relationships [23]. Therefore, a rich and complete ontology for internal logistics resources, especially focused on warehousing activities, is still lacking.

3. Research objectives

This paper proposes an extension for ontologies of the production systems to include a classification of internal logistics resources. In particular, the extensions have been based on the Manufacturing Systems Ontology (MSO), an ontology under development at Politecnico di Milano, within the European funded eScop (Embedded systems for Service-based control of Open Manufacturing and Process Automation) project [74].

So far, MSO has been proved as a useful model to describe mainly discrete manufacturing and process production systems [74–76]. The extension illustrated hereinafter is aimed at further detailing the internal logistics domain (i.e. material handling and warehousing) to the MSO by adding its specific concepts and relationships into the model. Specifically, the paper proposes a holistic taxonomy of internal logistics concepts and maps the relationships between them. The guiding idea has been to define concepts and relationships that are worth being included into an ontology that covers the internal logistics domain, thus addressing the identified gaps in the existing literature. Possible beneficial applications of an ontology into logistics systems are those defined in Sections 1 and 2.

This objective has been formulated into the following research questions (RQ):

- RQ1: ‘Which are the key concepts to be taken into account when defining an ontology for the internal logistics domain?’
- RQ2: ‘Which are the relationships among concepts to be modelled when defining an ontology for the internal logistics domain?’

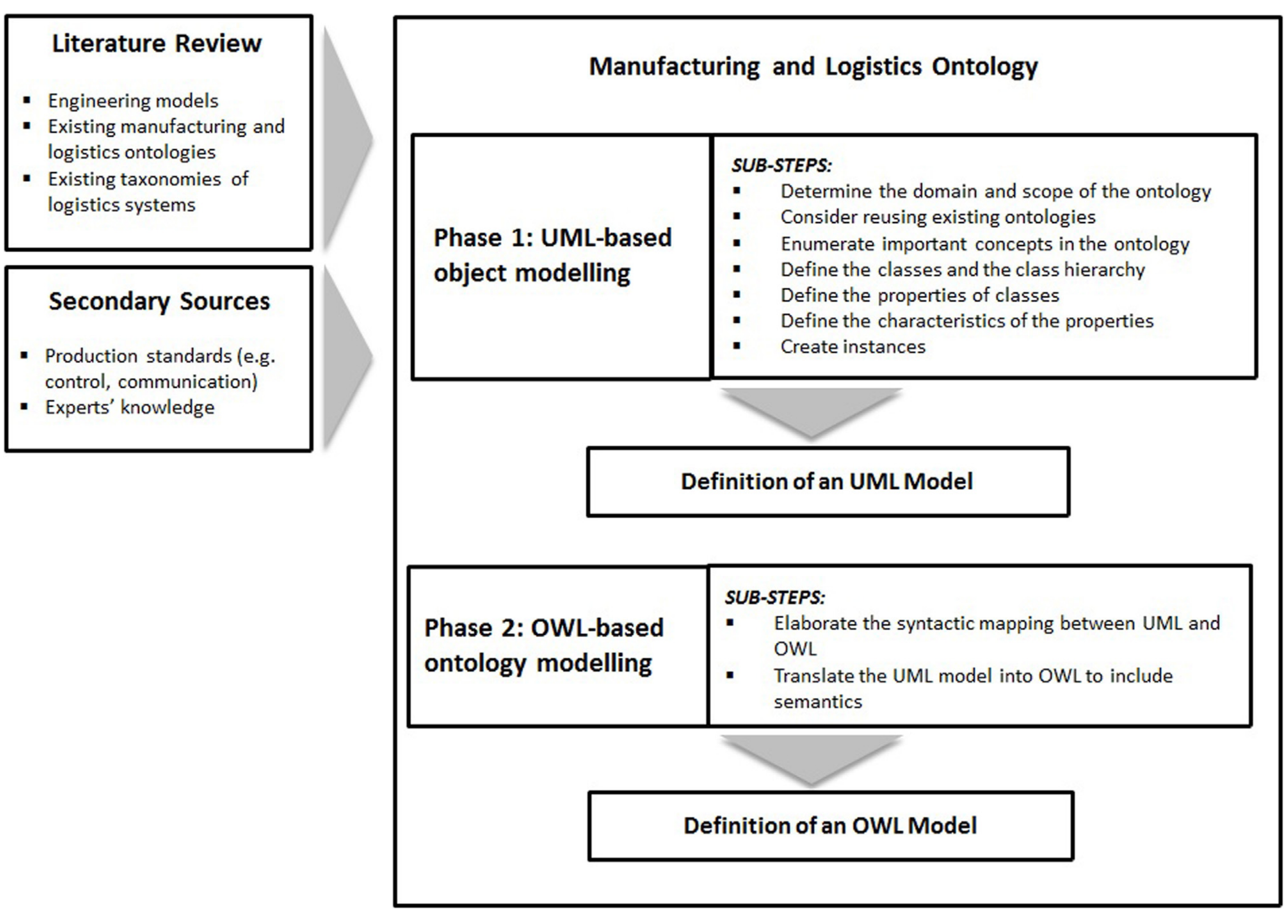


Fig. 1. Methodology adopted to build an ontology.

The remainder of the paper will: (i) briefly describe the main elements of the MSO ontology on which the extension for the internal logistics domain are meant to be contextualised; (ii) identify the internal logistics concepts that must be represented into the ontology, together with their logical characteristics, relationships, and hierarchical links.; (iii) provide Unified Modelling Language (UML) graphs to model concepts and relationships belonging to the internal logistics domain.

4. Research method

The adopted research method consists in an adaptation of the methodologies proposed by Lin et al. in 2011 and by Noy and McGuinness in 2001 [41,77]. The method allows to create a Web Ontology Language (OWL) model of the MSO and its internal logistics extension. The selection of the OWL language is based on previous research contributions that suggest it as the most suitable modelling language for the domain of interest [78,79].

The work presented in this paper is intended to map the domain knowledge into UML language, and to be used as a basis for the modelling activity into the more formal and machine-understandable OWL language. The choice of presenting the UML model of the domain as a basis for the following translation into OWL is motivated by the fact that the UML notation is easily understandable thanks to its visual nature [41]. For this reason, it is ideal as a notation for the engineers to express the domain concepts and relationships. Only in a second moment, knowledge engineers may take the UML model and build on it a semantic mapping. This is needed to have a translation between the UML model and the OWL. The UML language in fact is not proper for the ontological modelling of the internal logistics, as part of the domain of manufacturing systems, as suggested by [78], because it has no semantics properties and therefore no automated reasoning capabilities, that are instead offered by OWL [78]. In fact, being the OWL language formal and machine-understandable, it allows interoperability among systems and offers the potential for knowledge sharing and re-use. Moreover, using a semantic language such as OWL, it is possible to deduct new knowledge through reasoning and inference. In this way, it is possible to structure knowledge only partially in an explicit way (with classes and relationships), while part of it may remain implicit (as an example, with no explicit connection between classes) and be generated when required, following reasoning and inference techniques [14].

The need for adopting a two step-approach is already suggested by [80]: jumping from knowledge acquisition to implementation into a semantic language (coding an ontology) without a conceptual modelling phase in between would then require re-engineering when domain experts and system users are involved, to make the conceptual model explicit. Currently, OWL ontology editors offer a graphical function, that represents the selected classes and their hierarchy, offering to visualise diagrams of the classes and individuals connections that are expressed in OWL, without supporting the building of new parts of the model with a visual interface.

From [77], the methodology adopted to build the UML-based model has been composed of seven main steps, i.e. from the determination of the domain and scope of the ontology to the creation of instances. Fig. 1 represents the methodology to create first the MSO ontology and then to build the extension to the internal logistics domain.

The modelling process starts with the UML-based object modelling. This step needs previous material on the existing knowledge on the domain as input, such as engineering models, taxonomies, existing ontologies and standards and provides the UML conceptual model of the domain as output. It encompasses seven sub-steps:

- The first important decision to be made is the determination of the domain and scope of the ontology that is under construction;
- The second sub-step requires to consider the existing knowledge on the domain, such as the documents indicated as input to this step (i.e. engineering models, taxonomies, existing ontologies and standards);
- Third step is the enumeration of the important concepts in the domain;
- Starting from the important domain concepts identified at the third sub-step, classes and the class hierarchy is created as a fourth sub-step. For each concept, a class is created and it is indicated whether this class is a specialisation of another class inheriting all its characteristics, thus defining the hierarchy among the classes.
- In the fifth sub-step, the classes are enriched with properties, i.e. with attributes and relationships among them;
- The properties are characterised with constraints, ranges and other facets in the sixth sub-step;
- The last sub-step covers the creation of instances to represent a specific individual of the domain of interest. For the created ontology, the instantiation could cover a manufacturing system or a logistics system; with the creation of instances of the single components, their attributes and the relationships among them.

The second step is the OWL modelling of the domain ontology. It takes the UML model from the previous step as input and provides the OWL ontology as output. This is done through the two following sub-steps:

- Elaboration of the syntactic mapping between UML and OWL. In fact, as the two languages are different a mapping between the two is performed. This sub-step could be done in an automated way, as suggested by [41];
- The real translation from UML to OWL must be performed manually because the semantics, which is the additional characteristic of the OWL language, must be added to the syntactic mapping that does not include it.

The above-described methodology has been adopted to build the MSO ontology and its extension to the internal logistics domain. This paper presents the outcome of the first step for the extension: it presents the UML models for the internal logistics domain that includes the necessary concepts to be incorporated into the MSO. The work has been preceded by and based on the outcomes of an in-depth analysis of the domain (comprising the analysis of already defined taxonomies found in literature, previous knowledge suggested in literature, existing logistics domain ontologies and standards).

5. The Manufacturing Systems Ontology (MSO)

The Manufacturing Systems Ontology (MSO) is a structured representation of the domain of manufacturing systems, based on the object-oriented methodology. It enables the description of all the relevant aspects of a generic production system [75]. The modelling method defines a manufacturing system by addressing four different aspects separately, i.e.:

- the *physical aspect*, which contains the physical (static) definition of the system including workers, production facilities (such as tools, jigs and fixtures), material handling equipment and other supplementary devices;

- the *technological aspect*, which defines the transformational (i.e. functional) view of the system, considering the conversion process of the production factors;
- the *control aspect,* which stores data and describes the relationships among concepts that are needed to perform the production system control;
- the *visualisation aspect*, representing and storing data for the visualisation interfaces for human users, such as screens and visual devices.

The MSO comprises the four mentioned aspects, but for the scope of this paper only the physical aspect will be described, with specific concern to the internal logistics, while other aspects of the MSO are hereinafter neglected. Making references to the following documents for an in-depth presentation of the complete MSO model structure: [74–76].

5.1. MSO structure

The main concept on which the physical aspect is based on is the 'subsystem', i.e. an aggregation of resources as illustrated in Fig. 2. Subsystems can be composed either of smaller subsystems or of 'components', i.e. the elementary physical objects in the production or logistics system.

Components have an *ID* as attribute, i.e. a code that allows its unique identification in the system, and can be specialised in different types. Each of the following subclasses of components is linked with the 'father class' component with a hierarchical inheritance: they present the same attributes and relationships of the component class (Fig. 3). Specifically:

- *Processors* which perform transformation processes,
- *Transporters* which handle and transport workpieces or materials,
- *Storages* which stock workpieces and materials,
- *Unit Loads (ULs)* which are the basic handling units,
- *Tools* required for executing operations,
- *Fixtures* which hold tools,
- *Controllers* which are any decisional element performing functions of production planning and control in a manufacturing system,
- *Operators* who are people in the production system performing transport, processing, assembling, monitoring activities,
- *Sensors* which are devices whose purpose is to detect changes in an environment or to measure a certain physical or chemical variable.

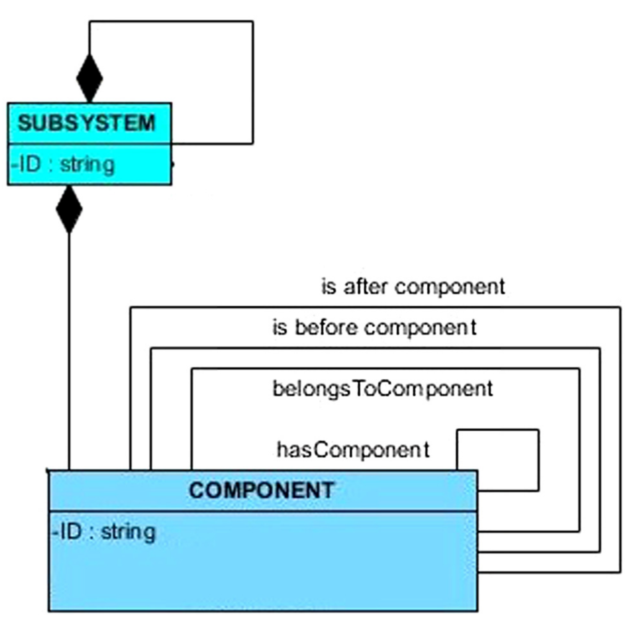


Fig. 2. The subsystem and the component classes.

Among components, those particularly related to the internal logistics domain are:

- *Unit Loads*: A unit load can be simply defined as the means used to move and handle one or more workpieces at one time. ULs are made in different sizes and from different materials determined by the size, weight, geometry, environmental requirements, etc. of the goods handled; they can be of different types: the most common in the manufacturing systems are bins, boxes, baskets, disposable and reusable pallets. They are designed to be compatible with the existing storage systems, transport systems and processors, and vice versa, therefore the ULs influence and are influenced by the choice of the type of transporter, the type of warehouse and the type of processor;
- *Transporters*: entities performing a transport function, i.e. moving material between different areas of the plant layout. Examples are automated guided vehicles (AGVs), conveyors, fork lift trucks, and other manual or automated transport machines, dedicated to the function of transferring and handling workpieces or materials to various locations throughout the factory;
- *Storages*: entities performing a storage function, i.e. keeping material for later use into the industrial process; examples are buffers, automated storage and retrieval systems (AS/RS), dedicated to the more or less temporary storage of the workpieces.

The subsequent sections will be dedicated to the latter two, respectively transporter class and storage class, proposing concepts that are necessary for an expansion of the MSO model to include the internal logistics domain.

In order to build a manufacturing and logistics system model, subsystems and elementary components must be connected with each other. In the proposed model, the connection is represented through the following types of relationship:

- Hierarchical relationship: represented by a line with a triangle pointing to the 'father class' and usually presenting the name: 'is a'. It indicates that the attributes and relationships of the father class are inherited to the other class, that is called sub-class (an example in Fig. 3 is that 'storage' is a subclass of 'component' and consequently will inherit the *ID* attribute and the relationships to other components such as 'belong to component' or 'is after component');
- Composition: represented as a line with a black diamond close to the class which is composed of the other class (as an example see Fig. 2: subsystem is composed of components);
- Generic association that can be named according to different needs: as an example the component class has four associations to itself: 'has component', 'belongs to component', 'is after component' and 'is before component' (e.g. Fig. 3). This means that any component (and, subsequently, any of the sub-classes of the component class) can be related to any other component or subclasses with a spatial relationship (is before or after in the plant) or with the indication that a component is mounted on another and belongs to it (or the reverse 'has it'). It is then possible to assert that a sensor belongs to a processor and that the processor has a sensor.

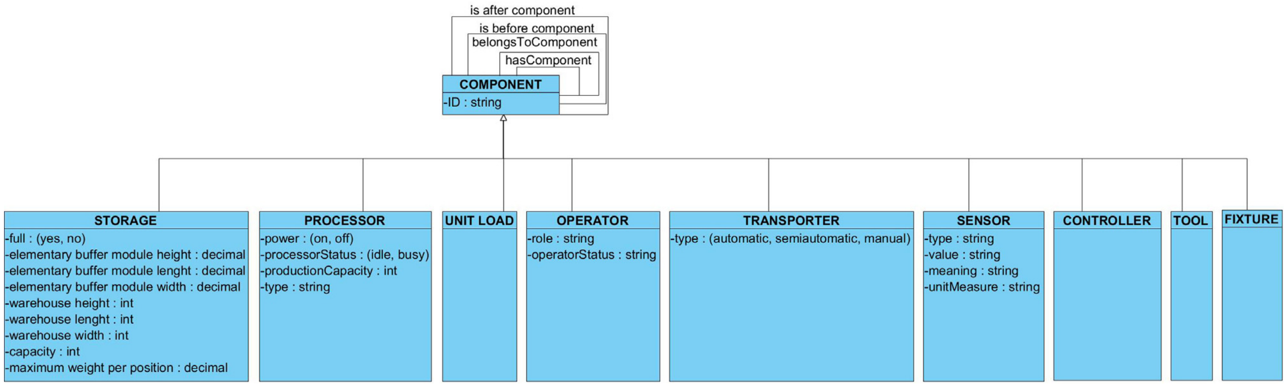


Fig. 3. Specialisations of the component class.

5.2. Definition of the 'transporter' class

The primary function of a transporter lies in the transfer and handling of workpieces or flows of material. Attributes of the transporter class are the *ID* attribute inherited from the component class, and its specific attribute *type*, which can either be:

- Automatic, when the transport activity is completely performed by a motorised machine, which doesn't need the continuous presence of an operator to run, whereas the only operator's activities may be supervision, programming and maintenance;
- Semi-automatic, when the transport activity is carried out by a physical device, which needs the continuous presence of an operator (for instance, a motorised device which must be driven by an operator, or a manually controlled crane, or a man-guided unpowered cart, or a stacker crane with a person aboard);
- Manual, when the transport activity is directly performed by a worker, without any physical device.

The transporter class is subdivided into:

- Discrete transporters, which perform independent transport missions. This implies that a single transport activity may be performed once the previous activity has been completed;
- Continuous transporters, which perform transport in a continuous stream, e.g. on a chain. This implies, that thanks to their structure, they can move a number of different workpieces simultaneously.

5.2.1. Discrete transporters

The discrete transporters inherit the attributes from the father classes 'component' and 'transporter'. They also present a specific attribute: *weight capacity per carrier,* which expresses the maximum load per carrier (kg/carrier).

Discrete transporters can be either 'area restricted' or 'vehicles'. The former ones are used to move and handle workpieces within a limited area or volume, while the latter are used to move workpieces along a path (i.e. fixed route versus not-fixed route).

Area restricted discrete transporters can be divided into:

- Cranes, i.e. automated overhead equipment to move loads intermittently within a limited area or volume;
- Handlers, i.e. devices equipped with a gripper. They can be characterised by the following attributes: *arm rotation speed* (°/min), *arm in/out speed* (m/min), *arm up/down speed* (m/min), *horizontal reach* (m), *vertical reach* (m), *maximum angle of rotation* (°). They consist in 'robots' (programmable, multifunctional units designed to move workpieces or specialised devices between set points in space via a continuous path, for performing a variety of tasks), that can be specialised according to the geometry of the positioning axis of the arm. The latter indicates the shape of the working volume, which is defined as the set of points that can be reached by the gripper of the arm (Cartesian robots, anthropomorphic robots).

Vehicles are characterised by the attributes: *speed* (m/min) which can be lifting or horizontal speed, and *acceleration* (m/min^2). They can be divided into:

- Guided path discrete transporters (GPDT), used to move workpieces along a predetermined path. The path serves as a guide for mobile units or load carriers with their own propulsions units. Usually the *type* attribute in GPDT is set to automatic. GPDT also can be characterised by their specific attributes: *route* which can be fixed or semifixed (the route is defined as fixed when the GPDT are used to move workpieces

Table 2
Summary of the specialisations of discrete transporters.

Class	Subclasses		Examples
Vehicles	Guided path vehicle (GPDT)	Non-lifting GPDT vehicle	Cart-on-track, Automated guided vehicle (AGV), Laser guided vehicle (LGV), Tow line conveyor, Rail mounted vehicle
		Lifting GPDT vehicle	Lift, Automated guided vehicle (AGV), Laser guided vehicle (LGV)
	Free path vehicle (FPDT)	Non-lifting FPDT vehicle	Order picking cart, Transpallet, Hand cart, Tractor trailer
		Lifting FPDT vehicle	Order picker truck, Counterbalanced truck, Stacker, Reach truck, Very Narrow aisle (VNA), Side Loading Forklift Truck
Area restricted discrete transporters	Cranes		Overhead travelling crane, Jib crane, Stacker crane, Traslo-elevator
	Handlers	Anthropomorphic robots	–
		Robots	Cartesian coordinates robot, Cylindrical coordinates robot, Polar coordinates robot

along a guided path, with a predetermined transport route without route alternatives; while it is semi-fixed when the GPDT are used to move workpieces along a guided path but with route alternatives), and *number of vehicles*. They are specialised according to their lifting capability, i.e. 'non-lifting GPDT' (vehicles provided with translational movement, but without any lifting movement – except a small lifting necessary for downloading-uploading the UL) and 'lifting GPDT' (with both translational and lifting movements).
- Free path discrete transporters (FPDT), not following any fixed path. Typically, FPDT have a semi-automatic *type*. They can be specialised according to their lifting capability, i.e. 'non-lifting FPDT' and 'lifting FPDT', with the same approach than GPDT. The FPDT are characterised by the following attributes, as examples: *maximum lift height* (m), *lifting weight capacity* (kg), *required minimum aisle width* (m), *turning radius* (m).

The mentioned specialisations in which discrete transporters are classified are reported in Table 2, that also presents the classification of the transporters of this type.

Discrete transporters have two characterising elements that are needed to be represented in a conceptual model of the domain: the 'path' concept and the 'location' concept.

The path is the set of points that a vehicle can or must follow. The path can be 'single' if it is without intersections, directly connecting two component of the manufacturing system; it can be either a line or a close loop; or 'combined' if it is composed of single paths. In general, the path can be curved or straight, with or without intersection, open or close loop. The path can be either a single track (i.e. in case of a path directly connecting two stations), or a combination of more tracks (i.e. in case of path with intersections); in this last case, the path is divided into single tracks linked between them. In case of a transporter serving more stations in line, the path is divided into single paths connecting the stations two by two; between each couple of tracks there is the location element, which models the stopping positions in front of the stations. The design of a GPDT involves both the definition of the vehicle itself and the definition of the path. In fact, the design of these transporters (i.e. an AGV) consists to a great extent in the definition of the path. The same approach can be followed in modelling the FPDT: in fact, in this case, even if there isn't any fixed path, the transporter is allowed to run along the areas of a plant which are not occupied by other physical resources. Therefore, after having designed the plant, it is possible to identify the feasible path that FPDT can follow; the feasible path specific of every transporter depends also on the service activities assigned to it. Therefore, also FPDT can be modelled by defining both the vehicles and the path along which they can move. In case of an elevator the path is vertical. The path concept has some characterising attributes, as examples: *capacity* (maximum number of vehicles that may be on the path at once) and *length* (m).

Locations can be for instance points in correspondence of which there is loading/unloading of workpieces to/from another component (a processor, a warehouse or another transporter) or the processing activities in case that workpieces are directly processed on the transporter holding them. This concept is important because once the path has been defined, it is important to define those points along the path in which a vehicle (belonging to the discrete transporters class) can stop (even if theoretically the vehicle is able to stop in any point). The location concept can be overlapping with the path concept when the vehicle moves along them without stopping, or it can be totally separated when there are stopping positions, where the vehicle stops. A location can have a single position (when there is a single stopping position in which the vehicle can stop) or multiple positions (when there are multiple stopping positions one next to another and the vehicle can stop in any of them). From this consideration, it can be stated that the connection between the two concepts (location and path) is that a single path can be either a *simple path* (if it doesn't have the stopping capability) or a *location* (if it allows vehicles to stop in that position).

5.2.2. Continuous transporters

Continuous transporters inherit the attributes from the father classes 'component' and 'transporter'. In particular, they are usually characterised by the value of the *type* attribute set to 'automatic'. Specific attributes of this class and its subclasses can be, for instance:

- *horizontal speed* (m/min): horizontal speed of the continuous transporter;
- *vertical speed* (m/min): lifting speed of the continuous transporter;
- *length* (m): it is the total length of the continuous transporter;
- *incline* (°);
- *acceleration* (m/min^2);
- *width* (mm): continuous transporter width (i.e. belt or cart width);
- *minimum feed lag*: minimum time lag between two subsequent loadings;
- *moving method*: powered or unpowered (examples of unpowered are the manual systems or systems by gravity);
- *minimum curve radius* (mm);
- *accumulation* (Boolean values: 'yes' or 'no'): continuous transporters can be able to accumulate workpieces; this attribute will be set to 'yes' if the continuous transporter is used for the accumulation of workpieces or to 'no' if not.

For some kinds of continuous transporters accumulation is easy to achieve (i.e. roller, chute and wheel conveyors), while for others it is very hard to achieve (i.e. chain, belt and slat conveyors). For these latter, accumulation is generally achieved at some points, by combining them with modules of accumulating conveyors (wheel or roller) for accumulation purposes, even though numerous other methods can be employed for this objective. Therefore, the accumulation capability depends on the physical characteristics of conveyors. Then the possibility for a conveyor to accumulate generally depends on the placement of specific systems, such as sensors and blockage systems. In fact, during the design process of a conveyor (i.e. a roller conveyor) the designer defines which modules of the conveyor must allow the accumulation and which modules must not. For this reason, continuous transporters that are not able to accumulate workpieces (i.e. chain, belt and slat) will have a 'no' value in the accumulation attribute, while continuous transporters able to accumulate workpieces (i.e. roller, chute and wheels) will have a 'yes' or 'no' value according to the specific design case. A continuous transporter accumulating workpieces can be considered both as a storage (since it is able to store a certain number of ULs and it is characterized by the attributes of storages, such as the size of the buffer, and implements a specific queue discipline) and to the transporter class.

Continuous transporters can be specialised in:

- Bulk loading continuous transporters (BLCT), where workpiece and material loading is continuous, therefore it is possible to load workpieces and materials in any point of the transporter. In BLCT the overall capacity depends on the size of workpieces. BLCT are usually characterised by a zero-value *minimum feed lag*. They also present the attribute of *linear weight capacity*, that expresses the load per linear meter (kg/m).
- Discrete loading continuous transporters (DLCT), where workpieces or materials are loaded in the transport system at one or

more predefined points or at a predefined distance from the previous workpiece or material unit. The attribute *minimum feed lag* in the case of DLCT must be fixed for each DLCT instance. Other specific attributes of this class can be the *weight capacity per carrier* (expressing the maximum load per carrier in kg/carrier) and the *number of carriers*.

Table 3 represents the list of possible transporter types belonging to the continuous transporters, divided into bulk and discrete loading.

BLCT are composed of a number of modules, that can be unidirectional, or with multiple inflows or outflows of workpieces. For this reason, it has been specified that they are in a relationship of 'composition' with:

- 'Unidirectional modules': a conveyor module that receives workpieces from a previous module or another device and is only able to make it proceed to the next module.
- 'Table modules': a conveyor module that has both the transportability of moving the workpiece the next module and receiving it from (or sending it to) different directions.

The subclasses of the "Transporter" class are represented graphically (according to the UML notation) in Fig. 4.

5.3. Definition of the 'storage' class

The primary function of storage in a production system is to house material for staging or building inventory for later use in the industrial process. The storage system is composed of elementary components (also called elementary buffer modules), each presenting geometrical and structural characteristics (maximum size and weight allowed), which together create the overall capacity of the total storage system.

Specific attributes of this class and its subclasses can be, for instance:

- *full* (Boolean: yes/no): it indicates if the storage space has ended (with "yes" value) or if there is still space ("no" value);
- *elementary buffer module height* (mm): height of the single elementary storage component;
- *elementary buffer module length* (mm): length of the single elementary storage component;
- *elementary buffer module width* (mm): width of the single elementary storage component;
- *warehouse height* (–): number of elementary buffer module in the height direction;
- *warehouse length* (–): number of elementary buffer module in the length direction;
- *warehouse width* (–): number of elementary buffer module in the width direction;
- *capacity* (–): number of ULs that the overall storage system is able to store;
- *maximum weight per position* (kg): maximum weight allowed per each UL that is stored into the storage system.

Storage systems can be classified into:

- Continuous Storage: storage types typical of the process industry (e.g. that can house liquids, for grains, for gases . . .);
- Discrete Storage: storage types typical of the discrete manufacturing (e.g. storage for single components, workpieces and finished products).

Among the discrete storage types, we can divide among:

- Single-depth storage, also called direct access, because all the stored ULs are directly accessible and it is therefore possible to retrieve any UL.
- Multiple-depth storage, also called indirect access, where the access is limited to some ULs and a specific discipline regulates the sequence of UL retrieval (LIFO – Last In First Out – or FIFO – First In First Out – disciplines);

Table 4 synthesizes the storage types and hierarchy.

The subclasses of the class storage are represented graphically (according to the UML notation) in Fig. 5.

6. Industrial example

An industrial example in the mechanical industry has been used to show how the modelling of the system can be applied to an internal logistics context. The examined system is composed of a miniload for case picking and storage operating within a factory warehouse. It is used for retrieving cases, which are the unit loads of this system, or single workpieces (i.e. single items contained in the cases) both to feed the production departments (i.e. an assembly station) and to serve final customers (i.e. picking station). Fig. 6 provides a schematic representation of the system under consideration. As an example, Fig. 7 reports a 3-D image of a miniload system with three aisles.

The miniload consists of two aisles characterised by single-deep, double-sided storage racks. Each storage position is the same size and can hold one case (cases are sized $400 \times 600 \times h = 300$ mm; max 20 kg). Each aisle has 27 storage levels (unit height clearance per storage position – including allowances – equal to 550 mm), with a storage capacity of 2.538 cases for each aisle, thus being the overall capacity equal to 5.076 cases. Each aisle is served by a crane (maximum horizontal velocity 6 m/s, maximum vertical velocity 3 m/s; acceleration/deceleration rate of 2,7 m/s^2). Cranes move horizontally and vertically simultaneously when picking or storing cases, guided by a control system. The input/output (I/O) point is located at one end of each storage aisle. One buffer (called buffer out) handles the cases which have been retrieved, the other one (called buffer in), located on the other side of the storage aisle, handles the cases to be stored.

The miniload is connected to a roller conveyor operating in a closed loop with automatic divert mechanisms and accumulation lanes. When performing storage activities, cases wait for the crane – whereas cases are taken from the buffer out and loaded on the roller conveyor in case of picking – so that roll conveyor and crane can work independently from one another. The conveyor feeds: (i) a picking station – also used as case loading/unloading bay (i.e. full cases coming from the receiving area/empty cases coming out of the miniload) – and (ii) an assembly station. Both picking and

Table 3
Continuous transporters.

Class	Examples
Bulk loading continuous transporters (BLCT)	Chute conveyor, Piping systems
Discrete loading continuous transporters (DLCT)	Belt conveyor, Slat conveyor, Tray conveyor, Trolley conveyor, Chain conveyor, Roller conveyor, Power-and-free conveyor, Wheel conveyor, Cross-belt

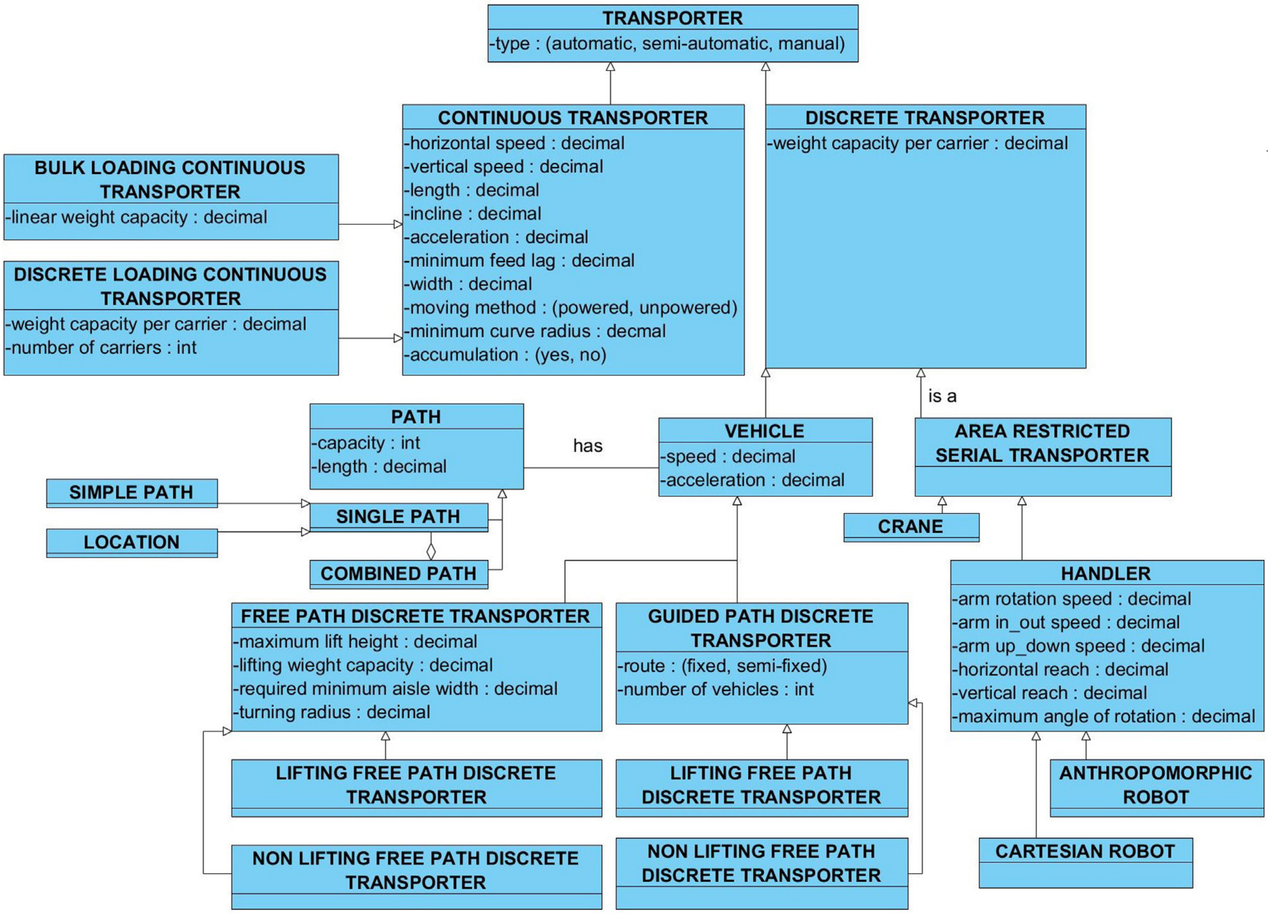


Fig. 4. Subclasses of the transporter class.

Table 4
Storage types.

Class		Subclasses		Examples
Storage	Continuous Storage	–		Tank, Silo
	Discrete Storage	Single-depth Storage		Single-deep selective rack, Bin shelving, Drawers in cabinet, Mobile storage, Vertical Storage System, Single-deep Miniload, Horizontal Carousel, Vertical Carousel, Single-deep AS/RS, Single-deep-AVS/RS
		Multiple-depth storage	FIFO	Flow Rack, Drive-through, Accumulating continuous loading parallel transporter, Multiple-deep AS/RS, Multiple-deep-AVS/RS
			LIFO	Multiple-deep rack, Block stacking, Push back rack, Drive-in rack, Stacking frame, Multiple-deep miniload, Multiple-deep AS/RS, Multiple-deep-AVS/RS, Pallet rack with Flow rail

assembly stations have buffers (a buffer in and a buffer out, respectively).

In case of a picking activity, the control system identifies the aisle in which the workpiece to be picked is located in the miniload, and guides the crane to the storage position. The crane retrieves the entire case and moves it to the roller conveyor. The control system determines the destination station for each retrieved case, i.e. picking station versus assembly station. After their arrival to the station (i.e. picking or assembly), operators select the required workpiece(s) from the case. In the picking station, the operator picks the required workpiece(s) based on the customer's order and places them into a box, to be transferred to the warehouse dispatch area, consolidated and finally shipped to their final destination. In case of an assembly operation, workpieces retrieved from the miniload are assembled into finished products, and these latter are loaded into cases that are eventually stored into the miniload.

The described industrial case can be modelled through the use of the classes and relationships that were defined in Section 5 of this paper. The overall system can be called "Miniload System" and is an individual of the "Subsystem class", with the type "miniload".

The Miniload System is composed of four Subsystems: the two aisles and the picking and assembly stations. These can also be considered as subsystems, as they are composed of modules.

The **Aisle 1** (attribute "type: aisle"), which is composed of a crane to move workpieces horizontally and vertically along the aisle, two single-deep miniload storages, a buffer to input workpieces and a buffer to output workpieces from the storages. They can be represented by creating individuals of subclasses of the Component class. In particular:

a Crane1 can be represented as an individual of the following class hierarchy: Component, Transporter, Discrete Transporter, Area

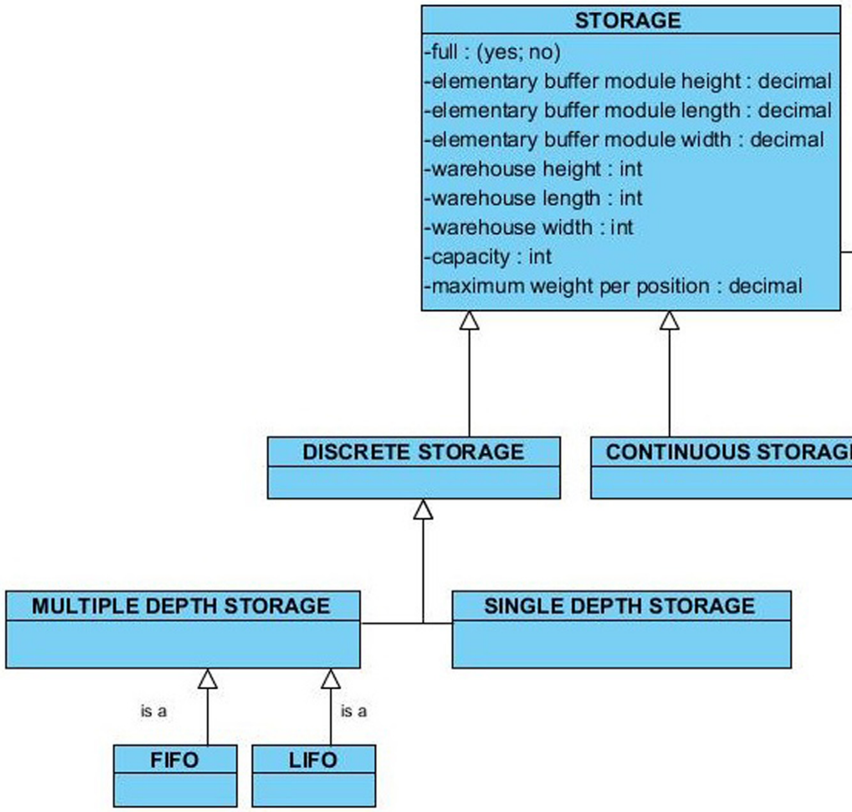


Fig. 5. Subclasses of the storage class.

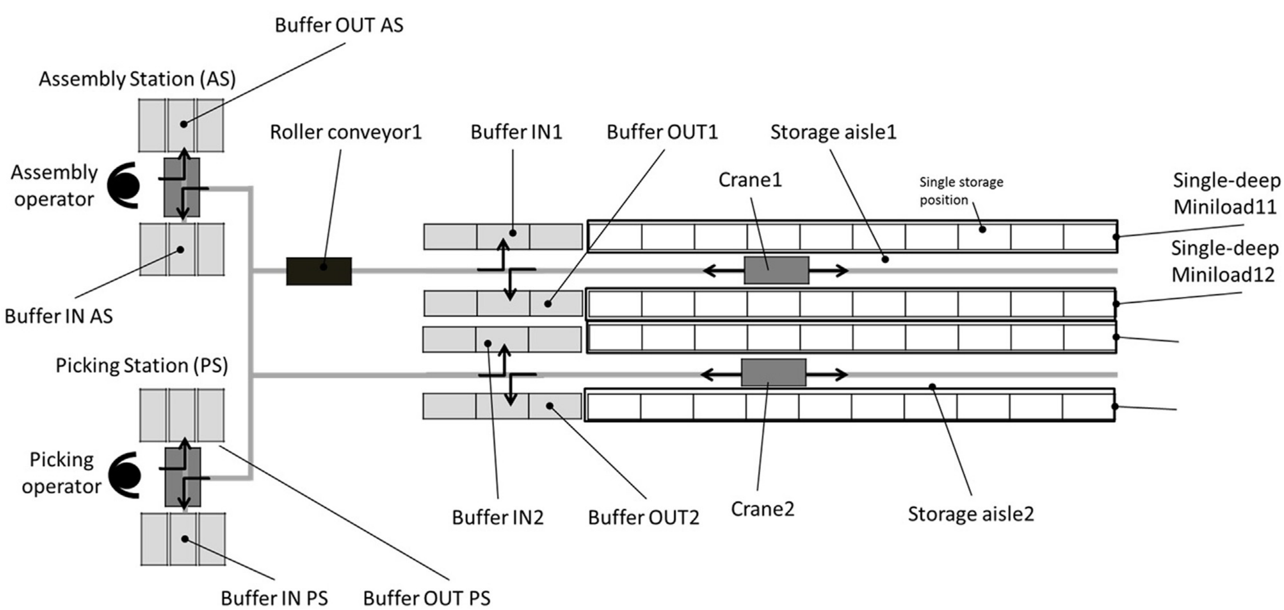


Fig. 6. Miniload system representation.

Restricted Discrete Transporter, Crane. Its attributes are representing its physical characteristics, according to the provided description of the case: horizontal and vertical speeds respectively equal to 6 and 3 m/s, acceleration equal to 2,7 m/s and moving method with “powered” value.

b Single-Deep Miniload 11 and Single-Deep Miniload 12 can be represented as two individuals of the following class hierarchy: Component, Storage, Discrete Storage, Single-Depth Storage, Single-Deep Miniload. Their attributes are representing the physical characteristics: geometry of the single storage module (elementary buffer module height equal to 300 mm, length equal to 400 mm, width equal to 600 mm), geometry of the miniload storage (warehouse height: 27 cases, length: 10 cases,

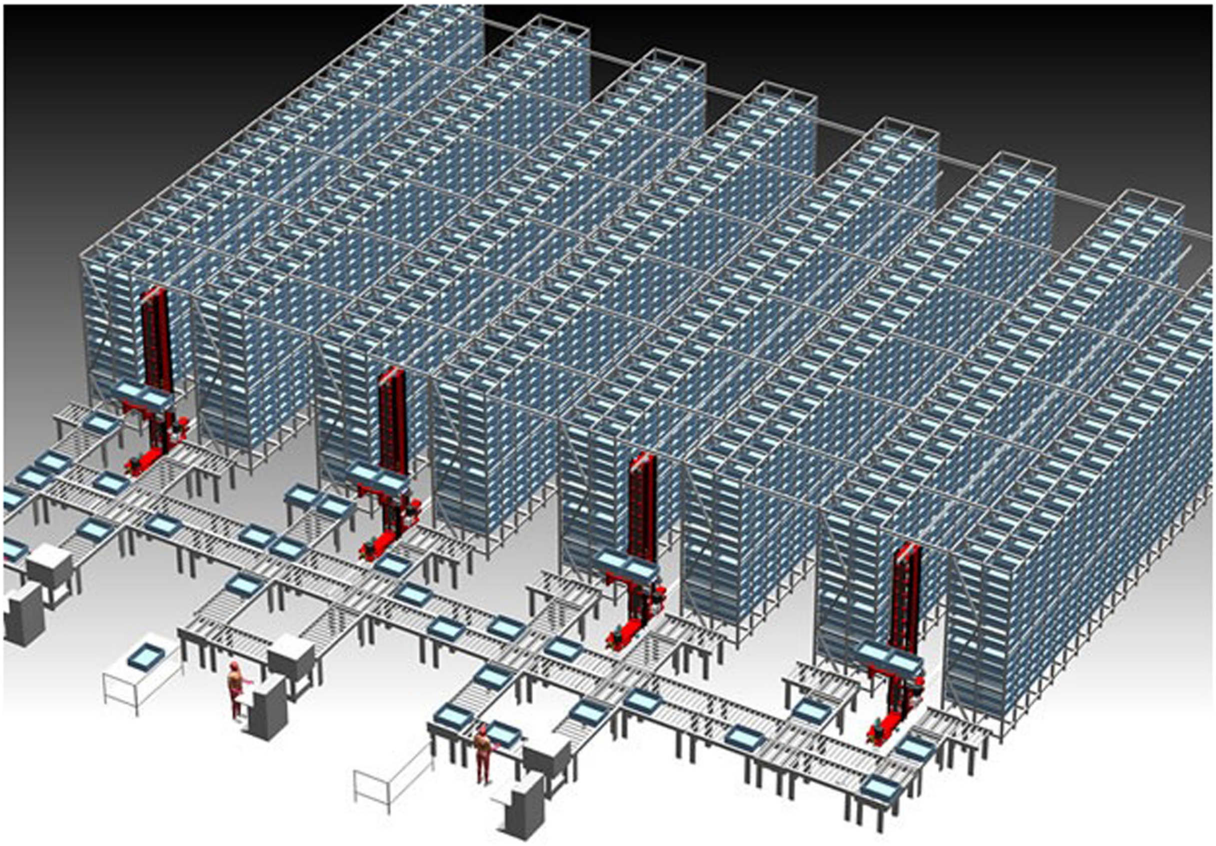


Fig. 7. Example of a miniload system.

width: 1 case), capacity equal to 1269 cases and weight allowance: maximum 20 kg per position.

c Buffer-In1 and Buffer-Out1 can be represented as two individuals of the following class hierarchy: Component, Storage, Discrete Storage, Single-Depth Storage. Their attributes are capacity of storing cases (equal to 3 cases, according to the case description).

2- The **Aisle 2** (attribute "type: aisle") is represented in the same way as the Aisle 1, and its components also have a direct correspondence to the already mentioned in Aisle 1, also because

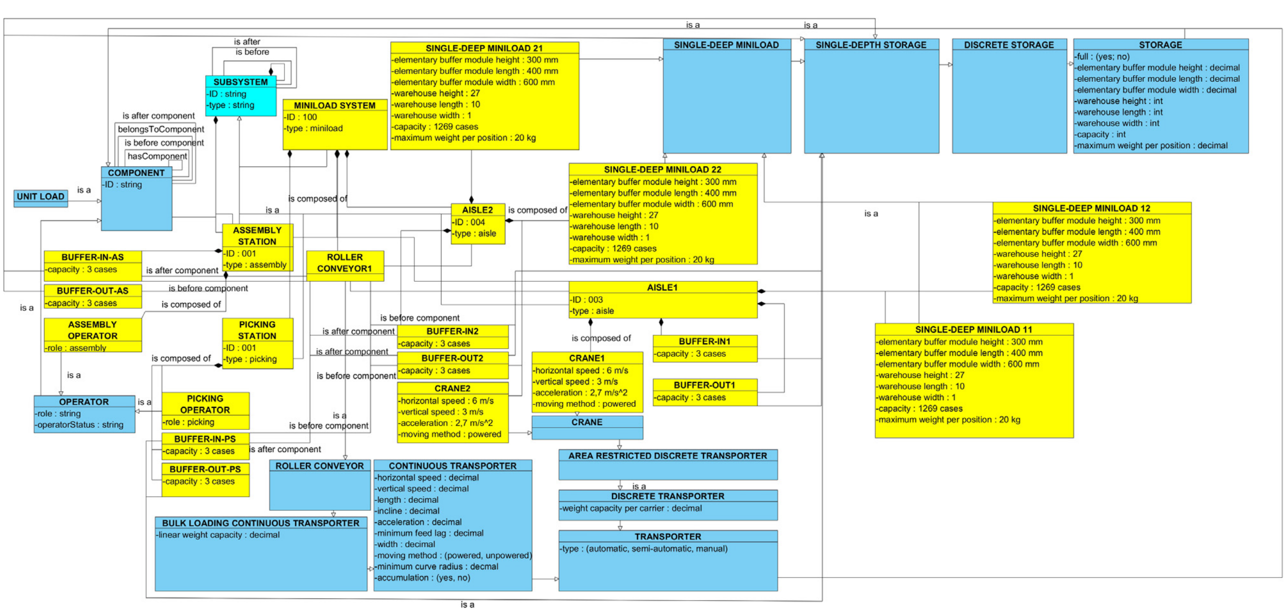


Fig. 8. UML instantiation of the logistics system.

their physical characteristics are the same, according to the case description, so also the attributes will receive the same values.

a Crane2: it has the same representation as Crane1, see description of Crane1.
b Single-Deep Miniload 21 and Single-Deep Miniload 22: they have the same representation of the Single-Deep Miniload 11 and Single-Deep Miniload 22, please see description above.
c Buffer-In2 and Buffer-Out2: they have the same representation of Buffer-In1 and Buffer-Out1, please see description above.

3- The **Picking Station** (attribute “type: picking”), which is composed of a buffer to input workpieces, a buffer to output workpieces from the station, and of a picking operator:

a Buffer-In-Ps and Buffer-Out-Ps can be represented as two individuals of the following class hierarchy: Component, Storage, Discrete Storage, Single-Depth Storage. Their attributes are capacity of storing cases (equal to 3 cases, according to case description).
b Picking Operator is an individual of the operator class and has as attribute is “role: picking”.

4- The **Assembly Station** (attribute “type: assembly”), which is composed of a buffer to input workpieces, a buffer to output workpieces from the station, and of an assembly operator:

a Buffer-In-As and Buffer-Out-As can be represented as two individuals of the following class hierarchy: Component, Storage, Discrete Storage, Single-Depth Storage. Their attributes are capacity of storing cases (equal to 3 cases, according to case description).
b Assembly Operator is an individual of the operator class and has as attribute is “role: assembly”.

The Miniload System is also composed of a Roller Conveyor, which is a component, as it is a basic module of the system. Its role is to connect the various parts of the system, thus allowing workpieces to be moved along the above-presented subsystems. For this reason, it has relationships “is after component” and “is before component” with the input buffers of the assembly and picking stations and with the input and output buffers of the aisles.

The graphic representation with UML notation of the considered miniload system is shown in Fig. 8. The domain knowledge structure is recalled in blue colour (for simplicity only the involved classes), while the individual instantiation representing the single object of the system is in yellow colour. Also the attributes have been given a value, according to the description of the above-reported case.

After the UML modelling, the suggested internal logistics system description must be translated into a semantic language in all its parts (classes, relationships, attributes) by instancing the OWL ontology structure, in order to be able to exploit the initial purposes and intended uses of the instanced ontology. As an example, reasoning and knowledge inference could be performed by automated software systems to make use of the explicit and implicit knowledge stored into the ontology. At this stage, the semantic mapping, the hierarchical relationships, the properties and the functional and symmetrical features of certain properties should be carefully dealt with in order to employ lower modelling effort, keep the ontology consistent and ensure the reasoning will produce consistent results and end within a limited time [14,78].

This type of representation could be the basis for different applications that may allow numerous technical and industrial benefits. From a technical perspective, the MSO ontology – of which the paper proposes an extension for internal logistics aspects – is intended to support the service- and CPS-based production systems orchestration, acting as a knowledge support to the Manufacturing Execution Systems (MES) and control system [81]. The ontology stores all information needed to control and orchestrate the system: it is then possible to automatically retrieve and update information in the ontology through services that produce queries reflecting the current status of the system. This enables to support the typical operations of a system: monitoring the production and logistics system, providing information on either the technological cycle or logistics routing that a workpiece shall follow in the system, the current status of resources (idle, working or faulty), or the main production Key Performance Indicators (KPIs) computed from sensor data stored in the ontology [82]. In particular, the proposed extension of the MSO is offering these potential in the internal logistics field [44]. Moreover, by offering a unique vocabulary to represent the internal logistics concepts, the proposed ontology could be used to ensure interoperability of the different devices and systems implemented in a logistics system. This leads to several benefits achievable from an industrial perspective: it supports re-configurability and flexibility of the system, as an up-to-date ontology can offer the required information (explicitly or implicitly through reasoning). A typical information offered by the ontology to the various services and devices in the system is about how they should collaborate, exploiting the technical potential for interoperability and support to control systems; in this way requiring less efforts of human manual programming, since the automated system could be automatically configured accordingly [6]. This opens the way to faster commissioning times for both new production and logistics systems and modifications of existing ones, leading to shorter time to market for new products.

7. Conclusions

Ontologies for production systems have received a strong research attention in the past years. They have been developed, proposed and applied to different contexts and scenarios and to different purposes (i.e. simulation, control, design, planning to name a few). Aspects of the production domain have not been treated equally by previous research contributions. In particular, many developments have been carried out in the field of discrete manufacturing, while logistics aspects have not received the same attention so far, despite their key role in industrial systems. Internal logistics, and in particular warehousing aspects (including storage, picking and handling activities), are usually not treated in the previous works on ontologies, with very few exceptions. However, ontologies representing the internal logistics field could be extremely valuable in today’s unpredictably-changing market, because they could act as the knowledge support to allow companies to achieve higher modularity and responsiveness. In fact, the current industrial situation is now facing a new era where modularity will play a key role and the challenge will be the effective deployment of modular solutions. Thanks to the ontology, a machine-readable description of each concept of the system and of the relationships between concepts is possible, thus supporting a full modularity and adaptability of the information systems connected to the production or logistics system. This description enables an easier and faster re-configurability of the system when needed.

The paper aims at offering a representation of logistics aspects, to be used as an extension to production systems ontologies that have been more focused on manufacturing processes so far. In particular, the authors envision this contribution to be inserted in an already developed ontology, the MSO, developed within the European funded project eScop. This representation reflects what has become a de-facto standard terminology in industry and among researchers in the field.

The scope of the modelling activity has been the logistics resources and the relationships among them. This has been motivated also by the fact that the existing literature on the topic usually focuses on the logistics processes, not modelling the resources that perform them.

The paper presents the structure of the hierarchical relationships among the internal logistics elements, such as Storage and Transporters, structuring them in a series of classes and subclasses, suggesting also the relationships and examples of attributes to complete the modelling.

Finally, the paper proposes an industrial example with a miniload system to show how such a modelling of internal logistics elements could be instanced into a real case.

The implications offered by this research work are both academic and practical. From an academic perspective, this work has two implications: on the one side, it offers an internal logistics classification distinguishing from the previous ones for the focus on the physical resources and elements (as well as relationships among them) rather than on logistics processes, as it appears from the literature analysis section of this paper. On the other side, this classification is at the basis of the creation of an internal logistics extension to an existing ontology. This contributes to the research stream of ontologies to support logistics activities, so far mainly focussed on supply chain issues rather than on internal logistics.

From a practical perspective, the technical and industrial benefits discussed in Section 6 have an impact on the economic and technical performances of the system. In fact, the faster ramp-up and commissioning times in case of new logistics systems enable new ways for manufacturing companies' responsiveness, based on the higher flexibility and re-configurability of semantic knowledge-based systems. From an economic point of view, this leads to both lower costs in terms of software development and shorter lead-times, improving the customer service and, thus, company profitability.

The described research work also paves the way to interesting future research paths, as it presents the description of the internal logistics field to be used as an extension to the MSO, the ontology for the manufacturing systems domain. This joint representation of the logistics and production areas is in line with the most advanced research trends in the context of smart manufacturing. In particular, the presented representation could be one of the enablers to the recently-formulated Synchro-push approach to production management, as described in [83]. Besides, it will be relevant to develop advanced maintenance systems featuring Prognostics and Health Management capabilities, combined with the modularization of supported functions and information to provide high flexibility for different applications in manufacturing [84,85]. Further research to better frame the role of manufacturing and logistics ontologies in production environments based on the Synchro-push paradigm is envisioned, in particular to understand the benefits brought by such an approach through its application to simulated and real industrial cases.

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*.

The authors E. Negri, S. Perotti, L. Fumagalli and G. Marchet, together with the colleagues of the Manufacturing Group at Politecnico di Milano, take the chance to express their gratitude to their colleague and mentor, Professor Marco Garetti, who recently passed away. He was the true inspirer of the presented research work. This paper represents his last published research work on an international peer-reviewed journal. Some of his last research ideas are further reported in his work about the Synchro-push paradigm referenced in the "Conclusions" section.

References

[1] Y. Koren, M. Shpitalni, Design of reconfigurable manufacturing systems, J. Manuf. Syst. 29 (2010) 130–141, doi:http://dx.doi.org/10.1016/j.jmsy.2011.01.001.

[2] Y. Alsafi, V. Vyatkin, Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing, Robot. Comput. Integr. Manuf. 26 (2010) 381–391, doi:http://dx.doi.org/10.1016/j.rcim.2009.12.001.

[3] J. Hoxha, A. Scheuermann, S. Bloehdorn, An approach to formal and semantic representation of logistics services, Proc. Work. Artif. Intell. Logist. (AILog), 19th Eur. Conf. Artif. Intell. (ECAI 2010), Lisbon, Port., 2010, pp. 73–78.

[4] A. Scheuermann, J. Hoxha, Ontologies for intelligent provision of logistics services, ICIW 2012 Seventh Int. Conf. Internet Web Appl. Serv. (2012) 106–111.

[5] M. Garetti, L. Fumagalli, Role of ontologies in open automation of manufacturing systems, Proc XVII Summer Sch. Ind. Mech. Plants – 12/9/2012-14/9/2012, Venice, Italy, 2012.

[6] J. Lastra, I. Delamer, Semantic web services in factory automation: fundamental insights and research roadmap, IEEE Trans. Ind. Inf. 2 (2006) 1–11 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1593597 (Accessed 2 October 2014)..

[7] F. Dallari, G. Marchet, M. Melacini, Design of order picking system, Int. J. Adv. Manuf. Technol. 42 (2009) 1–12, doi:http://dx.doi.org/10.1007/s00170-008-1571-9.

[8] G. Marchet, M. Melacini, S. Perotti, A model for design and performance estimation of pick- and-sort order picking systems, J. Manuf. Technol. Manage. 22 (2011) 261–282, doi:http://dx.doi.org/10.1108/17410381211267682.

[9] G. Marchet, M. Melacini, S. Perotti, Investigating order picking system adoption: a case-study-based approach, Int. J. Logist. Res. Appl. 18 (2015) 82–98, doi:http://dx.doi.org/10.1080/13675567.2014.945400.

[10] M. Goetschalckx, J. Ashayeri, Classification and design of order picking, Logist. World 2 (1989) 99–106, doi:http://dx.doi.org/10.1108/eb007469.

[11] R. de Koster, T. Le-Duc, K.J. Roodbergen, Design and control of warehouse order picking: a literature review, Eur. J. Oper. Res. 182 (2007) 481–501, doi:http://dx.doi.org/10.1016/j.ejor.2006.07.009.

[12] T. Gruber, Toward principles for the design of ontologies used for knowledge sharing, Int. J. Hum. Comput. Stud. 43 (1995) 907–928 http://www.sciencedirect.com/science/article/pii/S1071581985710816 (Accessed 29 September 2014)..

[13] B. Chandrasekaran, J.R. Josephson, V.R. Benjamins, What are ontologies, and why do we need them? IEEE Intell. Syst. 14 (1999) 20–26, doi:http://dx.doi.org/10.1109/5254.747902.

[14] F. Giunchiglia, F. Farazi, L. Tanca, R. De Virgilio, The semantic web languages, Semant. Web Inf. Manag., Springer-Verlag, Berlin, Heidelberg, 2010, pp. 25–38, doi:http://dx.doi.org/10.1007/978-3-642-04329-1.

[15] P.F. Patel-Schneider, I. Horrocks, A comparison of two modelling paradigms in the Semantic Web, Web Semant. 5 (2007) 240–250, doi:http://dx.doi.org/10.1016/j.websem.2007.09.004.

[16] N. Guarino, Formal ontology and information systems, Ontol. Inf. Syst. Proc. FOIS'98 (1998) 3–15.

[17] N.F. Noy, Semantic integration: a survey of ontology-based approaches, SIGMOD Rec. 33 (2004) 65–70, doi:http://dx.doi.org/10.1145/1041410.1041421.

[18] X.H. Wang, D.Q. Zhang, T. Gu, H.K. Pung, Ontology based context modeling and reasoning using OWL, Proc. Second IEEE Annu. Conf. Pervasive Comput. Commun. Work. (2004).

[19] C. Schlenoff, R. Ivester, D. Libes, P. Denno, S. Szykman, An Analysis of Existing Ontological Systems for Applications in Manufacturing and Healthcare, 1999, doi: 10.1017/S0890060400144038.

[20] M. Garetti, L. Fumagalli, P-PSO ontology for manufacturing systems, INCOM 2012 – Inf. Control Manuf. Conf. 14 (2012) 449–456 http://www.ifac-papersonline.net/Detailed/53645.html (Accessed 1 December 1 2014)..

[21] M. Colledani, W. Terkaj, T. Tolio, M. Tomasella, Development of a conceptual reference framework to manage manufacturing knowledge related to products, processes and production systems, in: A. Bernard, S. Tichkiewitch (Eds.), Methods Tools Eff. Knowl. Life-Cycle-Management, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 259–284, doi:http://dx.doi.org/10.1007/978-3-540-78431-9.

[22] V. Fortineau, T. Paviot, S. Lamouri, Improving the interoperability of industrial information systems with description logic-based models—the state of the art, Comput. Ind. 64 (2013) 363–375, doi:http://dx.doi.org/10.1016/j.compind.2013.01.001.

[23] A. Scheuermann, J. Leukel, Supply chain management ontology from an ontology engineering perspective, Comput. Ind. 65 (2014) 913–923, doi:http://dx.doi.org/10.1016/j.compind.2014.02.009.

[24] L. Ramos, Semantic Web for manufacturing, trends and open issues: toward a state of the art, Comput. Ind. Eng. 90 (2015) 444–460, doi:http://dx.doi.org/10.1016/j.cie.2015.10.013.

[25] M. Garetti, L. Fumagalli, A. Lobov, J.L. Martinez Lastra, Open automation of manufacturing systems through integration of ontology and web services, IFAC Proc. Vol. (2013) 198–203, doi:http://dx.doi.org/10.3182/20130619-3-ru-3018.00169.

[26] M. Loskyll, J. Schlick, S. Hodek, L. Ollinger, T. Gerber, B. Pirvu, Semantic service discovery and orchestration for manufacturing processes, Emerg. Technol. Fact. Autom. (ETFA), 2011 IEEE 16th Conf. (2011) http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6058988 (Accessed 1 December 2014)..

[27] D. Diep, C. Alexakos, T. Wagner, An ontology-based interoperability framework for distributed manufacturing control, Emerg. Technol. Fact. Autom. 2007. ETFA. IEEE Conf. On. (2007) 855–862 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4416869 (Accessed 1 December 2014)..

[28] A. Giovannini, A. Aubry, H. Panetto, M. Dassisti, H. El Haouzi, Ontology-based system for supporting manufacturing sustainability, Annu. Rev. Control 36 (2012) 309–317, doi:http://dx.doi.org/10.1016/j.arcontrol.2012.09.012.

[29] Z. Usman, R.I. Young, K. Case, J.A. Harding, A manufacturing foundation ontology for product lifecycle interoperability, Enterp. Interoperability IV, Mak. Internet Futur. Futur. Enterp., 2010, pp. 147–155.

[30] K.-H. Lee, Y.-S. Kang, Y.-H. Lee, Development of manufacturing ontology-based quality prediction framework and system: injection molding process, IE Interfaces 25 (2012) 40–51.

[31] M. Cai, W.Y. Zhang, 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 41 (2011) 574–582, doi:http://dx.doi.org/10.1109/tsmca.2010.2076395.

[32] H.K. Lin, J.A. Harding, A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration, Comput. Ind. 58 (2007) 428–437, doi:http://dx.doi.org/10.1016/j.compind.2006.09.015.

[33] R. Jardim-Goncalves, J. Sarraipa, C. Agostinho, H. Panetto, Knowledge framework for intelligent manufacturing systems, J. Intell. Manuf. 22 (2011) 725–735.

[34] H. Panetto, M. Dassisti, A. Tursi, ONTO-PDM: product-driven ONTOlogy for product data management interoperability within manufacturing process environment, Adv. Eng. Inf. 26 (2012) 334–348, doi:http://dx.doi.org/10.1016/j.aei.2011.12.002.

[35] H. Panetto, R. Jardim-Goncalves, A. Molina, Enterprise integration and networking: theory and practice, Annu. Rev. Control 36 (2012) 284–290, doi:http://dx.doi.org/10.1016/j.arcontrol.2012.09.009.

[36] R. Jardim-Goncalves, C. Coutinho, A. Cretan, C.F. da Silva, P. Ghodous, Collaborative negotiation for ontology-driven enterprise businesses, Comput. Ind. 65 (2014) 1232–1241, doi:http://dx.doi.org/10.1016/j.compind.2014.01.001.

[37] F. Ameri, C. Urbanovsky, C. McArthur, A systematic approach to developing ontologies for manufacturing service modeling, Proc. Work. Ontol. Semant. Web Manuf. (2012) 1–14.

[38] F.B. Vernadat, Technical, semantic and organizational issues of enterprise interoperability and networking, Annu. Rev. Control 34 (2010) 139–144, doi:http://dx.doi.org/10.3182/20090603-3-RU-2001.0579.

[39] D. Kiritsis, Closed-loop PLM for intelligent products in the era of the Internet of things, Comput. Des. 43 (2011) 479–501, doi:http://dx.doi.org/10.1016/j.cad.2010.03.002.

[40] A. Matsokis, D. Kiritsis, Ontology applications in PLM, Int. J. Prod. Lifecycle Manage. 5 (2011) 84–97 http://inderscience.metapress.com/index/Q4J4H65441555387.pdf (Accessed 5 October 2014)..

[41] L.F. Lin, W.Y. Zhang, Y.C. Lou, C.Y. Chu, M. Cai, Developing manufacturing ontologies for knowledge reuse in distributed manufacturing environment, Int. J. Prod. Res. 49 (2011) 343–359, doi:http://dx.doi.org/10.1080/00207540903349021.

[42] W.J. Verhagen, R. Curran, Ontological modelling of the aerospace composite manufacturing domain, Improv. Complex Syst. Today, Springer, London, 2011, pp. 215–222.

[43] M. Imran, B. Young, The application of common logic based formal ontologies to assembly knowledge sharing, J. Intell. Manuf. 26 (2013) 139–158, doi:http://dx.doi.org/10.1007/s10845-013-0768-4.

[44] N. Chungoora, R.I. Young, G. Gunendran, C. Palmer, Z. Usman, N.A. Anjum, A.-F. Cutting-Decelle, J.A. Harding, K. Case, A model-driven ontology approach for manufacturing system interoperability and knowledge sharing, Comput. Ind. 64 (2013) 392–401, doi:http://dx.doi.org/10.1016/j.compind.2013.01.003.

[45] W. Long, Research on development method of MES based on component and driven by ontology, J. Softw. 5 (2010) 1228–1235, doi:http://dx.doi.org/10.4304/jsw.5.11.1228-1235.

[46] C. Legat, C. Seitz, S. Lamparter, S. Feldmann, Semantics to the shop floor: towards ontology modularization and reuse in the automation domain, Proc. 19th IFAC World Congr. (2014) 3444–3449 http://www.researchgate.net/publication/261361140_Semantics_to_the_Shop_Floor_Towards_Ontology_Modularization_and_Reuse_in_the_Automation_Domain..

[47] A. Ferrándiz-Colmeiro, V. Gilart-Iglesias, F. Maciá-Pérez, Semantic processes modelling independent of manufacturing infrastructures, Proc. 15th IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA 2010 (2010), doi:http://dx.doi.org/10.1109/ETFA.2010.5641229.

[48] J. Leukel, S. Kirn, A supply chain management approach to logistics ontologies in information systems, W. Abramowicz, D. Fensel (Eds.), Bis 2008, Lnbip 7 (2008) 95–105.

[49] S. Lemaignan, A. Siadat, MASON: a proposal for an ontology of manufacturing domain, Proc. IEEE Work. Distrib. Intell. Syst. Collect. Intell. Its Appl. (2006) 6–11 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1633441 (Accessed 1 October 2014)..

[50] D. Nadoveza, D. Kiritsis, Ontology-based approach for context modeling in enterprise applications, Comput. Ind. 65 (2014) 1218–1231, doi:http://dx.doi.org/10.1016/j.compind.2014.07.007.

[51] M. Gonzalez-Rodriguez, A. Vennesland, S. Dalmolen, A common language for logistics services interoperability, IEEE Int. Conf. Adv. Logist. Transp. (2015) 70–75.

[52] Q. Yang, L.O.I. Lei, Z. Feng, Interval-gap-based logistics time ontology modeling and validation for decision support, Proc. 2013 Int. Conf. Mach. Learn. Cybern., Tianjin, 2013, pp. 87–91.

[53] O. Sakka, P.-A. Millet, V. Botta-Genoulaz, An ontological approach for strategic alignment: a supply chain operations reference case study, Int. J. Comput. Integr. Manuf. 24 (2011) 1022–1037, doi:http://dx.doi.org/10.1080/0951192X.2011.575798.

[54] T. Grubic, I. Fan, Supply chain ontology: review, analysis and synthesis, Comput. Ind. 61 (2010) 776–786, doi:http://dx.doi.org/10.1016/j.compind.2010.05.006.

[55] J.P. Van den Berg, W.H.M. Zijm, Models for warehouse management: classification and examples, Int. J. Prod. Econ. 59 (1999) 519–528, doi:http://dx.doi.org/10.1016/S0925-5273(98)00114-5.

[56] C.J. Matheus, M.M. Kokar, K. Baclawski, J.A. Letkowski, C. Call, M. Hinman, J. Salerno, D. Boulware, SAWA: an assistant for higher-level fusion and situation awareness, Def. Secur. Int. Soc. Opt. Photonics 5813 (2005) 75–85, doi:http://dx.doi.org/10.1117/12.604120.

[57] C.J. Matheus, K. Baclawski, M.M. Kokar, J.J. Letkowski, Using SWRL and OWL to capture domain knowledge for a situation awareness application applied to a supply logistics scenario, Rules Rule Markup Lang. Semant. Web, Springer, Berlin Heidelberg, 2005, pp. 130–144.

[58] M. Fayez, L. Rabelo, M. Mollaghasemi, Ontologies for supply chain simulation modeling, M.E. Kuhl, N.M. Steiger, F.B. Armstrong, J.A. Joines (Eds.), Proc. 2005 Winter Simul. Conf. (2005) 2364–2370.

[59] J. Himoff, G. Rzevski, P. Skobelev, Multi-agent logistics i-scheduler for road transportation, Aamas'06 (2006) 1514–1521.

[60] P. Lian, D. Park, Design of logistics ontology for semantic representing of situation in logistics Hyuk-Chul Kwon, Second Work. Digit. Media Its Appl. Museum Herit. (2007) 432–437, doi:http://dx.doi.org/10.1109/DMAMH.2007.52.

[61] C. Chandra, A. Tumanyan, Organization and problem ontology for supply chain information support system, Data Knowl. Eng. 61 (2007) 263–280, doi:http://dx.doi.org/10.1016/j.datak.2006.06.005.

[62] S. Gonnet, M. Vegetti, H. Leone, G. Henning, SCOntology: a formal approach toward a unified and integrated view of the supply chain, Adapt. Technol. Bus. Integr. Soc. Manag. Organ. Dimens., IGI Global, Hershey, PA, 2007, pp. 137–158.

[63] J. Ha, Y. Wei, Y. Jin, Logistics decision-making support system based on ontology process, Int. Symp. Comput. Intell. Des. (2008) 309–312, doi:http://dx.doi.org/10.1109/ISCID.2008.128.

[64] D.M. Gimenez, M. Vegetti, H.P. Leone, G.P. Henning, D.M. Gime, PRoduct ONTOlogy: defining product-related concepts for logistics planning activities, Comput. Ind. 59 (2008) 231–241, doi:http://dx.doi.org/10.1016/j.compind.2007.06.019.

[65] Y. Ye, D. Yang, Z. Jiang, L. Tong, An ontology-based architecture for implementing semantic integration of supply chain management, Int. J. Comput. Integr. Manuf. 21 (2007) 1–18, doi:http://dx.doi.org/10.1080/09511920601182225.

[66] D. Park, G. Lee, H. Kwon, Ontological approach to integration of event-centric logistics information into EPC network, IEEE Int. Work. Semant. Comput. Appl. (2008) 1–8, doi:http://dx.doi.org/10.1109/IWSCA.2008.30.

[67] J.-C. Tsou, The ontology of a supply chain model based on IDEF5 and Ontolingua, Prod. Plann. Control 19 (2008) 265–274, doi:http://dx.doi.org/10.1080/09537280801973034.

[68] G. Nie, M. Fu, H. Xia, A semantic mapping system based on e-commerce logistics ontology, World Congr. Softw. Eng. (2009) 133–136, doi:http://dx.doi.org/10.1109/WCSE.2009.242.

[69] X. Zhang, T. Tian, Logistics domain ontology model and its application, Adv. Mater. Res. 108–111 (2010) 1403–1408, doi:http://dx.doi.org/10.4028/www.scientific.net/AMR.108-111.1403.

[70] F. Bonfatti, P.D. Monari, L. Martinelli, Proposal for a logistic domain ontology, P. Cunningham, M. Cunningham (Eds.), IST-Africa 2010 Conf. Proceedings. IIMC Int. Inf. Manag. Corp. (2010) 1–8.

[71] Y.L. Chi, Rule-based ontological knowledge base for monitoring partners across supply networks, Expert Syst. Appl. 37 (2010) 1400–1407, doi:http://dx.doi.org/10.1016/j.eswa.2009.06.097.

[72] T. Grubic, I. Veza, B. Bilic, Integrating process and ontology to support supply chain modelling, Int. J. Comput. Integr. Manuf. 24 (2011) 847–863, doi:http://dx.doi.org/10.1109/I-ESA.2009.60.

[73] Y. Lu, H. Panetto, Y. Ni, X. Gu, Ontology alignment for networked enterprise information system interoperability in supply chain environment, Int. J. Comput. Integr. Manuf. 25 (2012) 1148–1159, doi:http://dx.doi.org/10.1080/0951192X.2012.681917.

[74] S. Strzelczak, P. Balda, M. Garetti, A. Lobov, Open Knowledge-driven Manufacturing & Logistics. The eScop Approach, Warsaw University of Technology Publishing House, Warsaw, 2015.

[75] E. Negri, L. Fumagalli, M. Macchi, M. Garetti, Ontology for service-based control of production systems, APMS 2015, Part II, IFIP AICT 460 (2015) 484–492 (http://www.springerlink.com/index/10.1007/978-1-4020-9783-6)..

[76] L. Fumagalli, S. Pala, M. Garetti, E. Negri, Ontology-based modeling of manufacturing and logistics systems for a new MES architecture, APMS 2014, IFIP Adv. Inf. Commun. Technol. 438 (PART I) (2014) 192–200.

[77] N. Noy, D. McGuinness, Ontology development 101: A guide to creating your first ontology, (2001) pp. 1–25. http://liris.cnrs.fr/~amille/enseignements/Ecole_Centrale/What is an ontology and why we need it.htm (Accessed 3 October 2014).

[78] E. Negri, L. Fumagalli, M. Garetti, L. Tanca, Requirements and languages for the semantic representation of manufacturing systems, Comput. Ind. 81 (2016) 55–66, doi:http://dx.doi.org/10.1016/j.compind.2015.10.009.

[79] E. Negri, L. Fumagalli, M. Garetti, L. Tanca, A review of semantic languages for the conceptual modelling of the manufacturing domain, Proc. XIX Summerschool Ind. Mech. Plants Francesco Turco, Ancona, Italy, 9th–12th Sept. 2014, 2014.

[80] M. Fernández-López, A. Gómez-Pérez, J.P. Sierra, Building a chemical ontology using methontology and the ontology design environment, IEEE Intell. Syst. 14 (1999) 37–46.

[81] M. Garetti, L. Fumagalli, E. Negri, Role of ontologies for CPS implementation in manufacturing, MPER—Manage. Prod. Eng. Rev. 6 (2015) 26–32, doi:http://dx.doi.org/10.1515/mper-2015-0033.

[82] E. Negri, L. Fumagalli, M. Garetti, Approach for the use of ontologies for KPI calculation in the manufacturing domain, Proc. XX Summerschool Ind. Mech. Plants Francesco Turco, Napoli, Italy, 16th–18th Sept. 2015, 2015, pp. 30–36.

[83] M. Garetti, M. Macchi, A. Pozzetti, L. Fumagalli, E. Negri, Synchro-push: a new production control paradigm, 21st Summer Sch. Fr. Turco 2016 (2016) 150–155.

[84] A.J. Guillén, A. Crespo, M. Macchi, J. Gómez, On the role of Prognostics and Health Management in advanced maintenance systems, Prod. Plann. Control 27 (2016) 991–1004, doi:http://dx.doi.org/10.1080/09537287.2016.1171920.

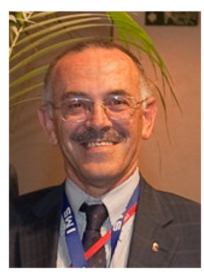
[85] L. Fumagalli, M. Macchi, Integrating maintenance within the production process through a flexible E-maintenance platform, IFAC-PapersOnLine, Elsevier Ltd., 2015, pp. 1457–1462, doi:http://dx.doi.org/10.1016/j.ifacol.2015.06.292.

Elisa Negri received her PhD at the Department of Management, Economics and Industrial Engineering of the Politecnico di Milano 2016, with a thesis about the role of ontologies for smart manufacturing. Her research fields are ontology-based engineering, industrial systems modeling, digital twin and Cyber-Physical Systems supporting advanced manufacturing solutions in the frame of Industry 4.0. She graduated in 2013 in Management Engineering at Politecnico di Milano with a MSc thesis developed in Aachen, Germany, at the FIR Institute for Operations Management. She received a double BSc degree in Mechanical and Production Engineering, by Politecnico di Milano and by Tongji University in Shanghai (PRC), with a focus on production processes.

Sara Perotti is an Assistant Professor at Politecnico di Milano (Italy), Department of Management, Economics and Industrial Engineering. She gained her M.Sc. in Mechanical Engineering, major in Transportation with highest honours at Politecnico di Milano. In 2010 she gained her PhD at Politecnico di Milano, where she currently lectures and undertakes research on logistics and transportation. Her present research fields are logistics and warehousing, supply chain sustainability, and information and communication technology (ICT) for freight transportation. She is a member of the Editorial Advisory Board of the *International Journal of Physical Distribution & Logistics Management*, and she reviewer for several scientific peer-reviewed international journals. She is author of a number of publications, including contributions in the *International Journal of Physical Distribution & Logistics Management*, *International Journal of Logistics Research and Applications*, *International Journal of Production Research*, *International Journal of Advanced Manufacturing Technology*, and *Journal of Cleaner Production*.

Luca Fumagalli is Assistant Professor at Politecnico di Milano. He got Master of Science in Mechanical Engineer in 2006 and then he has collaborated with the Department of Management, Economics and Industrial Engineering (DIG) of Politecnico di Milano since January 2007. In 2010 he completed the PhD program with a thesis about Innovation in Maintenance Management, related with the use of Ontologies. He works on different research topics about production management, industrial services and in particular maintenance management related topics, with a specific concern on new technological solutions. His research activity has been related also with European research funded projects. Luca Fumagalli is also presently responsible of the research of the observatory TeSeM (www.tesem.net) and vice-director of Master Megmi (Master Executive in Gestione della Manutenzione Industriale – Executive Master on Industrial Maintenance Management) delivered by MIP – School of Management – Politecnico di Milano and SdM – School of Management Università degli Studi di Bergamo (http://www.mip.polimi.it/megmi).

Gino Marchet is a Full Professor at Politecnico di Milano (Italy), Department of Management, Economics and Industrial Engineering. He currently lectures on Logistics and Industrial Plants, and is Director of the 'Material Handling' executive course at Politecnico di Milano Business School. At Politecnico di Milano he is also Director of the Material Handling Observatory since 2003, and of the Contract Logistics Observatory since 2011. Member of the Editorial Board of Logistica Management since 1990, i.e. the official journal of the Italian Logistics Association (AILOG), he is also a member of the Editorial Advisory Board of the *International Journal of Physical Distribution & Logistics Management*. He has written and contributed to over 100 textbooks and academic papers in the field of logistics. Gino Marchet can be contacted at: gino.marchet@polimi.it.



Marco Garetti was Full Professor of Industrial Technology at Politecnico di Milano and director of the Executive Master on Management of Industrial Maintenance. Born in Como on 1946, he got a master degree in Mechanical Engineering at the Politecnico di Milano in 1971. He started working in the R&D department of Alfa Romeo cars, then joined Politecnico di Milano in 1974, where he was founder of the University Degree in Logistics and Production Engineering in Lecco campus, after being director of the School of Production Organization since 1990. In the course of his career, he taught several courses, in the field of Asset Lifecycle Management and Design and Management of Production Systems. He was director of the Executive Master on Management of Industrial Maintenance in the School of Management of Politecnico di Milano. His research was based on maintenance management, sustainable manufacturing, manufacturing automation, production management and product lifecycle management. He was project leader in research projects on domestic and world-wide scale. Furthermore, he has been the organizer of many international conferences and has been author or co-author of 8 books and more than 200 papers. He also delivered training courses and performed consulting activity for private and public companies and institutions.