Towards a role-centric and context-aware information distribution system for manufacturing

Kosmas Alexopoulos¹, Sotiris Makris¹, Vangelis Xanthakis², Konstantinos Sipsas², Aggelos Liapis³, George Chryssolouris⁴, *

¹Laboratory for Manufacturing Systems and Automation (LMS), University of Patras, 26500 Patras, Greece
²INTRASOFT International SA, Research & Innovation Department

* Corresponding author. Tel.: +30 2610 997262; fax: +30 2610 997744. E-mail address: xrisol@lms.mech.upatras.gr

1. Introduction

The manufacture of products in the modern industrial world requires the combined and coordinated efforts of people, machinery, and equipment. Thus, a manufacturing system can be defined as a combination of humans, machinery and equipment that are bound by a common material and information flow [1]. Manufacturing information systems lie in the core of modern manufacturing automation systems, and play a key role for the optimization of the entire manufacturing operation. They are primarily responsible for manufacturing information processing, logistics management, monitoring and control of manufacturing process, production planning and production scheduling. The continuously increasing complexity and amount of information in manufacturing, are major challenges both for big and for Small and Medium Enterprises (SMEs). Nowadays, the manufacturing Information and Communication Technology (ICT) platforms provide a number of useful tools, such as sensors, wireless networks, mobile devices, Manufacturing Execution System (MES), Enterprise Resource Planning (ERP) and information management systems and they support shop-floor and back-office personnel in a series of activities. ICT solutions originally intended for the support of production processes, currently lack in providing the right information to the right person, at the right time and location. The enormous amount of information gathered and generated by IT systems needs to be presented in a manner that could truly speed up production processes, enable immediate reaction to issues and shortcomings. This requirement results in several technical shortcomings to be overcome by the ICT domain besides providing support to manufacturing companies.

- In many cases, information infrastructure fails to successfully aggregate and manage data from factory-wide sensor networks as well as from various data sources,
namely MES or ERP, analyze the data and deliver it to different users in a context-based manner.

- Information distribution usually neglects different user roles that may change according to the context the user operates in the production environment. Consequently, the information source is not aware of the information receiver’s role and thus, fails in delivering the required subset of information.

- For the delivery of personalized data and services to people in a production facility, the cognitive domains such as information retrieval, decision making, and situation awareness are required. For example, the cognitive load, especially of shop-floor operators, working under harsh conditions, is not properly addressed during the distribution of information.

Figure 1 shows the situation for providing information available in typical legacy systems currently installed in production. This approach considers only a static pre-defined description of a user’s role and a static definition of his context in order to deliver production related information. The approach followed in this paper, proposes that context and roles definition be considered during the information delivery between the factories’ legacy/ICT systems, such as MES and ERP systems and the people working in the shop-floor.

In this paper, there is a description of a conceptual framework along with its major modules that can facilitate the requirement stated above. The remaining of this paper is organized as follows. In chapter 2, a relevant literature review takes place. In chapter 3, there is a detailed presentation of the proposed context model and architecture and the main components of the context aware information system architecture. In chapter 3, there is a presentation of the way that this framework can be applied to a motivating industrial scenario. In chapter 5, there is a conclusion of the main findings and future research steps are proposed.

2. State of the art

In general, context-awareness is the property for the provision of suitable services to a user through the analysis of his context. The context can be any information characterizing the situation of an entity. The entity can be a person, place, or object relevant to the interaction between the user and a context-aware application. Such applications are capable of reacting specifically to their current location, time and other environment attributes and may adapt their behavior without explicit user intervention and thus, aim at increasing usability and effectiveness. In manufacturing, context aware applications define a holistic and dynamic context model that takes into account the context of tools, machines, parts, products, while at the same time, utilizes information regarding the planning of manufacturing processes. Context-awareness can be used to increase the visibility of operations and their performance in a manufacturing environment. It aims at enabling factory shop-floor and office personnel to make a decision, based on a systematic understanding of the system having derived from a real-time sensed context of the factory, instead of a decision based on a fragmentary system view and a limited expert knowledge. In this study, context-awareness is perceived as an information system’s capability to provide relevant information and services to the shop-floor user. The relevancy depends on the user’s task and the means of supporting decisions. These may be taken by shop-floor personnel by enabling advance perception and comprehension of tasks (e.g. assembly, maintenance, supervision) allocated to them. Potential errors that may occur during the production process can be avoided through proactive delivery of expert knowledge. In the design of context aware systems, there are two important design principles that should be considered: architecture and context model [2]. The architecture of a context aware system depends on different requirements, such as the location of sensors, the number of possible users, response times (real time, soft real-time, pull driven) or the availability of computing power. More insight into alternative architectures is provided in [2] and [3]. However, in many cases the layered architecture approach presented in Figure 2 is used, since it separates the detection of the context to its processing.

Apart from the architecture of a context aware system, the context data modeling approach is also a very important aspect. A popular way of classifying context instances is the distinction of the context dimensions to external and internal [4].

![Figure 2: Layered conceptual framework for context-aware systems](image-url)

In a factory environment, the external content can be distinguished into three entities: places (e.g. work station, sub-assembly line, assembly line, plant, and offices),...
people/roles (operators, support personnel, supervisors) and things (machines, tools, conveyors etc.). External context factors may be sensed through the off-the-shelf sensor technology (e.g. RFID, location technology, machine status). Each of these external entities may be described by various attributes, which can be classified into four categories: identity (each entity has a unique identifier), location (an operator’s position, co-location, proximity etc.), value (or activity, meaning the intrinsic properties of an entity, e.g., machine green/yellow/red status etc.) and time (used for timestamps to accurately define situation, ordering events etc.). Although the external dimension of the manufacturing context is important and very useful for context-aware applications, it should not be the only dimension of context that needs to be captured and used in a manufacturing environment. The internal dimension of context – user’s goals, tasks, work context, business processes, communication, emotional and physical state – also needs to be captured. For example, cognitive load, especially of shop-floor operators, working under harsh conditions, should be considered during the information distribution. In order to provide support through ICT systems in a context-aware manner in a manufacturing shop-floor, the cognitive domains such as situation awareness, decision making capability and work content are critical and have been given less attention by the scientific literature. However, internal context, such as task and cognitive domain aspects are difficult to be sensed. The work content of an operator could be identified by aggregating different pieces of information from different sources and information relevant to the external context should be available. Besides defining from what sort of entities and data the manufacturing context consists of a method for modeling the context should also be selected. A context data model as a fundamental element of the smart factory paradigm is presented in [5]. While in [6], context-awareness is proposed as an approach to providing additional information in support of the understanding of energy consumption in manufacturing environments.

In [7], there is a presentation of the most relevant context modelling approaches, which are based on the data structures used for representing and exchanging contextual. Early approaches include key-value and markup models. Key-value models use simple key-value pairs to define the list of attributes and their values describing context information used by context-aware applications. Markup-based context information models use a variety of markup languages including XML. The main critics of these approaches concern their limited capabilities in: (i) capturing a variety of context types, (ii) capturing relationships, dependencies, timeliness, and quality of context information, (iii) allowing consistency checking, and (iv) support reasoning on context, on context uncertainty and on higher context abstractions (see [8]). Graphical and object oriented models, such as the Unified Modeling Language (UML) has a strong graphical component (UML diagrams) and therefore it is also appropriate to modelling the context. The details of context processing are encapsulated at an object level and hence are hidden to other components. Access to contextual information is provided through specified interfaces. Logic based models have a high degree of formality. In a logic based context model, the context is defined as facts, expressions and rules. A logic base system is then used for adding, updating or removing new facts and thus for updating the context. The inference mechanism can be used in order to derive new facts, based on the existing rules of the logic based system. Finally, ontologies can be used to represent contextual information due to their high and formal expressiveness and the possibility to applying ontology reasoning. There are several context-aware frameworks that use ontologies as underlying context models. In [9], an ontology based approach is used for the development of a context repository for a manufacturing environment. However, the study focuses on enabling self-adaptation of flexible manufacturing processes, based on a context aware approach and not on information distribution in a manufacturing environment. In a similar manner in [10], an ontology-based context model also for real time decision making is proposed to optimize the key performance indicators of flexible manufacturing systems (FMS).

In this study, the ontology approach has been chosen for the modeling of the context data, mainly due to its superiority in expressing relations and dependencies among context data, heterogeneity and interoperability. However, the use of this technology brings for consideration some drawbacks such as timeliness and little support for modeling temporal aspects in ontologies as well as performance issues especially when applying inference in the ontological data.

3. The context-aware role centric approach for information distribution in manufacturing

3.1 Extended context data model for manufacturing

For the integration of context information from different context models an extended context schema is required. In this study, the ontology based approach is proposed for context data modeling. Ontologies allow context-modeling at a semantic level, to establish a common understanding of terms and enable context sharing, logic inference, reasoning and reuse. Ontology enables high and formal expressiveness of contextual information. Shareable ontologies are a fundamental precondition for knowledge reuse, serving as a means of integrating problem-solving domain-representation, and knowledge-acquisition modules. Several semantic specification languages, such as Resource Description Framework (RDF) and Web Ontology Language (OWL) provide potential solutions for context modeling. These standard formats for data-sharing, span across applications, enterprise and community boundaries. Users, both humans and machines, can share and understand the information available on the semantic web [10].

The proposed extended ontology schema consists of three parts:

- Top-level context ontologies: This part contains objects describing attributes, such as location, status or time. The top level context model consists of three modules:
- Events Ontology: This part contains entities describing events occurring in the manufacturing domain. An event is “something that happened” in the shop-floor and may contain attributes such as identity, location, value and time. The events are the basic elements for the inference of the context.

- Temporal Ontology: Contains entities describing the temporal nature of the context. A property or a relation in the ontology is valid within a time frame. The temporal dimension of a manufacturing context is relevant, since in some cases the content to be delivered to the users may depend on historical data and not only on the current, active, context. For example, assessing the skills of a specific role for the execution of a task may depend on the role’s frequency in this task during the previous months.

- Spatial Ontology: Contains classes describing the spatial/location entities. The definition of a spatial ontology is not in the scope of this manuscript.

Relations among the ontologies could be possible. For example, the position property of a spatial entity could have a temporal representation and an event may have a location property.

- Manufacturing domain ontology: In order for the status of a current factory to be presented, the ontology model has to include information on the resources (machines, tools, etc.) along with embed sensors and actuators, the products to be produced, production schedule and orders, the technical and organizational processes.

- Roles ontology: A schema describes the roles, played by manufacturing domain entities while performing activities. This schema can be used for allocating information services to shop-floor personnel as well as for authentication and authorization purposes.

In Figure 3, the relations among the main packages of the context data model for manufacturing are presented. The arrows in Figure 3 represent dependencies and relations among the components. For example, an entity in the Manufacturing domain model may have an object property, defined in the Temporal ontology, or an event entity in the manufacturing ontology may be a subclass of another entity in the Events Ontology.

**Figure 3:** The extended context data model for manufacturing

**Events model**
The purpose of the events model is two-fold. On the one side it provides the basis for storing into a knowledge base all the sensed events and on the other hand, the events will be used for the updating of the context. The study presented hereafter focuses on the second part, namely the development of an events model that can be used for the updating of the context model. The events are described by a layered and modular event ontology model. A shared top ontology for “raw” events will capture the top concepts, namely event timestamp and links to relevant event reader and assets (see Figure 4). Specific manufacturing domain ontologies, which are modular sub-ontologies of the common top ontology, then represent more concrete specialized concepts. The domain-specific events are modelled as domain classes of the manufacturing domain ontology and are in relationships with other classes of the manufacturing ontology. Then, a number of specific manufacturing shop-floor related event types are developed representing more concrete events.

As described in the chapter entitled “System’s architecture”, during the pre-processing phase, concrete manufacturing domain events can be generated from simple raw events.

**Temporal data model**
A number of different approaches for the representation of time in RDF have been proposed. These can be divided into three broad classes [11]: (a) the versioning approaches that maintain distinct RDF graphs representing the state of the domain at different points of time, (b) the abstract syntax extension approaches extending the RDF abstract syntax to incorporate notions of time and define corresponding semantics and (c) the conventional RDF approaches working within the current RDF syntax and semantics and model the changing state of the domain explicitly as triples in the RDF graph, perhaps using N-ary relations. In this study, of interest is the time dimension of the data, mainly for treating them as historical ones. Here, the N-ary approach has been selected mainly due to its direct implementation into semantic repositories. In Figure 5 the RDF representation of temporal attributes and object relations is presented by the example of modeling historical data for jobs’ execution on workstation and tasks’ execution on some resource (i.e. machine).

**Manufacturing domain model**
The current ontology aims at covering the manufacturing domain, at modeling the structure and the relationships among the primary physical as well as the virtual entities of a manufacturing system. The purpose of this ontology is to represent the plant, the product, the orders and the manufacturing attributes besides defining the interconnections. The proposed context ontology is used to defining a fundamental data model for context extraction in a manufacturing environment. The manufacturing ontology developed in this study utilizes the work presented in [13]. The overall ontology scheme consists of three main hierarchical structures with their interconnections: the resource hierarchy, product hierarchy and production orders structure. In the current ontology scheme (Figure 6), a coherent approach for resource hierarchy, following a four-level hierarchy [1], has been adopted and includes: plant level, job shop/flow line level, work center level and finally, the resource/machine level. The plant is the highest level in the hierarchy and corresponds to the system as a whole. The resource is regarded as a generic entity that can be a machine, a human worker or a storage area. A generic and abstract structure of a product is proposed for the needs of the
ontology. The main class of course is the MfgProduct. This concept represents the actual goods, produced. A product may be of some MfgProductModel and consists of a number of MfgPart. The production orders are broken down into MfgJob, which in turn, consist of a number of manufacturing tasks. An order corresponds to some part of the production facility (e.g. plant, flow-line). A job consists of tasks that can be released to some resource only. The ontology is defined as a generic manufacturing environment context model and a sector-specific context model. The generic context model consists of the generic hierarchical structures defined above. Then, a sector specific context model defines the concept such as: ProductA, PartA1, LoadPartToPallet, UnloadPartFromPallet etc.

Roles model

In a typical production facility, the tasks are assigned to different user roles, ranging from those of the operator to equipment engineer, process engineer, quality engineer, production planner, shift manager up to the production line manager. To support all those different roles the ICT systems need to fulfill dissimilar requirements. While the provision of only general information is not sufficient the full information creates information overload. The information source needs to be aware of the role of the information receiver and automatically delivers the required subset. The right information finally enables the instant and correct reaction to different situations at the production. In order for these difficulties to be overcome, context aware information should be combined with users’ role and users’ cognitive capabilities at any given time. The main purpose of having different roles is to organize the information distribution and access control services. In industrial practice, the various roles of an enterprise are usually interested in different types of information or expect a different view of the information. For example, workers on the shop floor are usually interested in the information, concerning material location and machine status in the working site. A line manager will be more interested in the site manufacturing status of the overall production line. In the same manner, different roles may have diverse privileges for accessing resources and devices. Moreover, mapping of information view and resource access to different roles should not be statically defined, but it could change dynamically based on the context that a user, being a member of the role, is active. For example, a member of the team or a line supervisor may have to view a specific workstation status information when located close to a machine, and an aggregated overview of the line when located away from the line. Similarly, a team supervisor could have the role of an operator, if for example, one is located close to a machine and no operator is sensed close to that machine, which is nevertheless operating normally. Such an approach could lead to context-based roles, as for an example TeamLeaderOnOperatorDuty that may provide the team leader with a specific view and access to workstation status data and instructions set. In this study, we model roles through a hierarchical approach. At the top level, the domain roles are defined while for each one of the domain roles a number of context based sub-roles are also being defined. The following are typical domain roles found in a manufacturing shop-floor:

- **Operator/worker** is a human resource, whose main task is to operate a particular work station or device and report to the supervisor the progress and problems made on it.
- **Team leader or team supervisor**, whose main task is to oversee the shop floor/assembly line and help resolve issues related to machine and human resources on the shop floor/assembly line in order for the desired production (production output, quality) and business set objectives (e.g. safety) to be met. Among others, he manages a shift of operators including training, balancing of personnel and may perform minor maintenance operations. The team supervisor is responsible for a part of a line and for reporting to the production line manager. He is usually a senior worker with recognized practice and knowledge on a specific task.
- **Production line manager** is in charge of an entire line and all its shifts.
- **Plant manager** has an overall responsibility and overview of the activities in the plant, usually through the ERP system.

![Figure 4: Events model](image1)

![Figure 5: N-ary relational approach to represent properties that vary over time](image2)
- **Warehouse operator** picks up the materials according to the orders received and loads them into pallets or similar material collecting equipment.
- **Logistics operator** collects all the loaded pallets from the warehouse and delivers them, usually with the use of forklifts, to the work stations at the shop floor/assembly line.
- **Maintenance operators** are notified of any malfunctions of machines, conveyor belts, and material handling equipment.
- **Engineering department** personnel update or prepare new designs, drawings and process steps for products and product variants.
- **IT department** personnel are responsible for maintaining the IT infrastructure, configure and update ERP/MES and sensors system.

The concept of the role in a context-aware information system, in manufacturing, is an important aspect since it can be used for decoupling the user (i.e. employee) and the context information. The system, in several occasions, will be more aware of the role than the user will and thus, it becomes less likely to misuse the context information (for instance location) against the workers. In a manufacturing shop-floor environment, a user should not be explicitly asked to log on to a system using the typical login, password boxes. This may not be efficient, especially when accessing a resource in a time critical task. Instead, sensing devices could identify the user’s role and maintain an implicit session. For example, an authorized worker can switch off a machine without needing to log off. The user’s role can be distinguished by his/her ID card. For instance, the user’s role can be stored into Near Field Communication (NFC) on a mobile device, his location at the shop-floor and his role or personal settings of the digital environment can be determined. According to this information, authentication, authorization and an improved information aggregation of data can be achieved.

### 3.2 System’s architecture

In this study, the generic layered architecture in Figure 2 has been instantiated in order to fit into the manufacturing, shop-floor requirements. The detailed framework discussed in this chapter is based upon the layered architecture in Figure 7. Each layer contributes to increasing visibility and context-awareness in a manufacturing environment by gradually enabling the easy sharing of various context information and sensor data, so that they can be used to complement each other and cover a factory-wide application environment. The layers are described in detail below:

- **Metrology and shop-floor sensor layer**: The shop-floor sensor data layer contains the physical hardware (sensors) and the adequate software (e.g. firmware). The sensor data are initially available as "raw" data. Then the "raw" data go through the first processing stage in order to improve their representation. The Metrology layer interprets the "raw" data into initial context information, and then stores it if required. For example, the sensor tracked signals are translated into valid shop-floor coordinates.

- **Preprocessing**: This module is responsible for the preprocessing of basic sensor events, before the identification of the context. An important functionality will be the transformation of the basic sensor events to manufacturing environment ones that are suitable for context identification. Moreover, this layer is used for the identification of critical events in order to quickly identify urgent situations. This layer utilizes the Complex Event Processing (CEP) approach to process, close to real-time events from the Metrology layer. Moreover, CEP provides the definition and construction of complex events from simple ones besides being an efficient way of recognizing complex event patterns and triggering alerts when time critical situations occur.

- **Data aggregation and context awareness**: In data aggregation layer context, information produced by different shop-floor sources and the Preprocessing layer is aggregated and a database holding the relevant information (real-time information, roles repository and historical data) is maintained. The Context Builder & Server module is responsible for building and updating the context data and provides services to it. In the context
awareness layer, high-level manufacturing activities context inference takes place based on the aggregated context information.

- **Context Builder and Server:** A generic solution for the provision of context based services to shop-floor personnel is provided in Figure 8 and can be applied to different manufacturing systems. The Context Builder module is responsible for identifying changes in the context and updating the context repository. The Context Builder uses all data provided by the events pre-processing layer as well as by other data sources (such as ERP and MES) to derive the current context. The data from the data sources is evaluated as for example in a typical situation, an event that states the arrival of a part to a workstation will be used to update the relations between the workstation object and the part it processes or to update the activity performed by the operator working in the workstation. Apart from the dynamic and continuous context updating, the Context Builder provides functionality for static context definition such as workstations and productions units. As shown in Figure 8, the Context builder mainly comprises two important modules, namely the Context Identification and Context Inference, which are described in detail shortly hereafter.

- **Context Identification:** This component is responsible for the identification of the current context, from events and other information sources, the ontology and the context information stored into the context repository. Identifying changes in the context (e.g. Line Supervisor is working as workstation operator) from sensed data is important for context-awareness and adaptation to the user's context changes.

- **Context Inference:** In many cases, context identification cannot be achieved directly from sensor data. However, new knowledge can be achieved by utilizing primitive events data. The inference can be made with the help of sophisticated reasoning techniques mainly relying on context representation. Rule inference engines can be used in context reasoning. Such reasoning systems from forward-chaining inference inherit the power of inferring knowledge (i.e. logical consequences) from sensed data and from the backward-chaining inference inherit the power of recognizing relevant context. Knowledge might also be deduced using an ontology inference facility.

- **Context Repository:** Inside the Context Repository, the identified context will be stored for further processing and reuse. An RDF store such as the Apache Jena [14] semantic repository is used for facilitating the need for storing the context.

The Context Server module not only does it provide access through queries to the context data but also listener/notification functionality so as for client services and applications to be notified of any specific types of changes in the context data.

- **Shop-floor applications:** In this layer, different context-enabled applications are deployed and provided to the final users (operators, production engineers, maintenance engineers and others) through different display devices. In factory environments, various visualization device types can be utilized to display and collect production-related data and information. The proposed architecture is generic enough to allow for different applications to utilize this infrastructure and provide specific end-user services. Typical aspects of these applications that can be deployed in the proposed framework can be used for example, to provide assembly instructions to an operator, on the basis of the product and workstation he is working on or to provide production schedule information based on the production line and the user’s role.

![Figure 7: Architecture of the context aware information handling and distribution in manufacturing.](image)

**Figure 7:** Architecture of the context aware information handling and distribution in manufacturing.

![Figure 8: Context builder and server architecture](image)

**Figure 8:** Context builder and server architecture

4. A context aware service for workstation instructions delivery

The capability of the proposed system in support of shop-floor users is demonstrated through an application for assembly operators. In this paragraph, the way that the proposed framework can be used in order to support operators, during assembly task execution in discrete manufacturing, is presented. Umberto, an inexperienced assembly operator, works in a plant that produces washing machines. The factory has a number of washing machine assembly lines and each line is structured in a number of sequential sub-assembly lines namely: the washing group assembly, the cabinet, the washing group assembly and the final packaging. As his
working shift is about to start, Umberto approaches the second work station, located in the washing group sub-assembly line. He has a mobile device, equipped with NFC and WiFi technology. The NFC tag stores role and profile data and by the time Umberto places his device close to an NFC reader, located in the workstation area, the sensor and metrology sub-systems identify the appearance of an inexperienced operator and the context database is updated through the context builder module. The production schedule data delivery service is notified by the context server and is adapted to the specific work station, that of Umberto’s language preferences and his experience, and presents an overview of the daily production schedule on the 17” fixed ruggedized touch screen that is located in the workstation. During operation, an RFID reader, located in the first workstation, reads the product information, stored in the RFID tag attached to the pallet, carries the part to be processed through the sub-assembly line, and the context builder, following the preprocessing phase, identifies a batch change in the line (i.e. the sensed part belongs to a new production batch). Moreover, the context repository “knows” the fact that the operations performed in Umberto’s work station are prone to introducing errors to this product variant. The Context Inference derives as a conclusion the situation of an inexperienced user working on an error prone assembly operation and the assembly task instructions delivery service is notified about the current situation. Thus, the instructions delivery service may adapt its content and graphical user interface in order for a detailed description of the workstation assembly instructions to be delivered to Umberto.

5. Summary and outlook

The study reported in this paper presents the conceptual framework of an information system that aims at achieving context-awareness to the dynamic operating environment of a manufacturing system. An RDF-store knowledge-based approach has been utilized for the collection of contextual entities, from several factory data sources, and represents them through an extended context model for further processing. The extended manufacturing context model as well as the context management approach provides a common interface for context acquisition, reuse and updates, which may be utilized by applications in support of shop-floor personnel in a manufacturing environment. High-level context is extracted from external data sources such as sensors and MES or ERP systems. The proposed framework of a context-aware information distribution system is based on a bottom-up approach for the development of such applications. A practical example from a discrete manufacturing industry is used in order to demonstrate the way that role and context depended information distribution services can be applied to a manufacturing floor. Future research will evaluate the system’s performance when processing a big amount of data. In that perspective, different data technologies such as distributed processing and map-reduce will be considered. Moreover, future research will focus on the development of algorithms that may use the proposed context model and the system architecture for automatic and semi-automatic optimization of the production system and the deployment of decision support clients in the shop-floor. In addition, the ontology based context model will be improved by future research to allow for reuse of this model on other types of industries as well as for application domains, mainly maintenance support. Finally, in order for the consideration of context in information distribution to be expanded, user’s internal context attributes such as cognitive load, should also be considered.

Acknowledgements

This research has been partially supported by the project ‘Sense&React – The context-aware and user-centric information distribution system for manufacturing’ funded by the European Union Seventh Framework Programme (FP7) under grant agreement n° 314350.

References

[1] Chryssoulis, G., Manufacturing Systems: Theory and Practice, 2nd Edition, Springer-Verlag, New York, New York, (2006)
[2] Baldauf M., Dustdar, S.,and Rosenberg, F., A survey on context-aware systems, Int. J. Ad Hoc and Ubiquitous Computing, 2007, Vol. 2, No. 4, pp. 263-277
[3] Hong J., Sub, H., Kim, S-J., Context-aware systems: A literature review and classification, Expert Systems with Applications 36 (2009), 8509–8522
[4] Prekop, P. and Burnett, M. (2003) ‘Activities, context and ubiquitous computing’, Special Issue on Ubiquitous Computing Computer Communications, Vol. 26, No. 11, pp.1168–1176.
[5] Lucke, D., Constantinouc, C., Weytkamp, W., Context Data Model, the Backbone of a Smart Factory, 42nd CIRP Conference on Manufacturing Systems, Grenoble, June 3-5, 2009.
[6] Herrmann, C., et. al., Context-Aware Analysis Approach to Enhance Industrial Smart Metering, Proceedings of the 18th CIRP International Conference on Life Cycle Engineering, Braunschweig, Germany, May 2 – 4, 2011.
[7] Strang, T., Limhoffs-Popien, C., A Context Modeling Survey, Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England
[8] Bettini C., et. al., A survey of context modelling and reasoning techniques, Pervasive and Mobile Computing, 6 (2010),pp. 161-180
[9] Scholze, S., Stokie, D.; Barata, J., Decker, C., Context Extraction for Self-Learning Production Systems, 10th IEEE International Conference on Industrial Informatics, INDIN, 2012.
[10] Uddin, M., et. al., Ontology-based context-sensitive computing for FMS optimization, Assembly Automation, 32/2 (2012) 163–174
[11] Gutierrez, C., Hurtado, C., Vaisman, A. , Temporal RDF , 2nd. European Semantic Web Conference (ESWC2005), May 2005, Heraklion, Greece.
[12] Hong J., Sub, E., Kim, S., Context-aware systems: A literature review and classification, Expert Systems with Applications 36 (2009), 8509–8522
[13] K. Efthymiou, K. Sipsas, D. Melekos, K. Georgoulas, G. Chryssoulis, "A Manufacturing Ontology Following Performance Indicators Approach", (DEIT 2011), ISBN 978-960-88104-2-6, 7th International Conference on Digital Enterprise Technology, Athens, Greece, pp. 586-595 (2011)
[14] Apache Jena, http://jena.apache.org/, accessed online November 2013.