

An Intelligent Context-aware System for Logistics Asset Supervision Service

Fan Feng, Yusong Pang and Gabriel Lodewijks
 Section of Transport Engineering and Logistics
 Delft University of Technology
 The Netherlands
 Email: {f.feng, y.pang, g.lodewijks}@tudelft.nl

Abstract—The use of Information and Communication Technology (ICT) has touched various aspects in the domain of transport engineering and logistics (TEL). As the development of TEL tends to be more complex in operation and large in scale, recent practices start to pay more attentions on improving system robustness and reliability. In addition, current ICT innovations (such as WSN and IOT) could record and deliver system descriptors (physical measurements, virtual resources, operational configurations) in real time. Such large-stream and heterogeneous data requires an integrated framework to process and management. To address such challenges, in this paper, a novel concept of context-aware supervision is proposed. An intelligent system with integration of semantic web and agent technology is proposed to support the concept realization, which aims at providing condition-monitoring and maintenance service to relevant user. A generic ontology-agent based framework will be illustrated. Finally, it will be applied for the supervision of a large-scale material handling system- belt conveying system as a proof-of-concept.

Index Terms—Context-awareness, ontology-agent integration, system supervision, material handling system

I. INTRODUCTION

THE USE of information communication technology (ICT) in TEL domain can be traced back to 1960s. It is chosen as a primary enabler to deal with increasing complexity of TEL development and enhance its competitive position with cost reduction and service promotion. Several conceptual ideas have been proposed that support the process of logistics and ICT technologies integration. The concept of *Integrated Logistics* is proposed to integrate IT with logistics management system to achieve synergy [1]. Afterwards, the concept of *E-Logistics* emerges that integrates Internet and mobile technology with logistics for providing one-stop value-added services to end-users [2], [3]. Recently, the concept of *Prognostics Logistics* has been put forward which utilizes wireless sensors (particularly RFID) together with decision making tools to enhance traceability and reliability of the logistics system [4].

Since the scale and complexity of TEL system has tremendously expanded, the attention of researchers and engineers have been shifted from enhancing operational efficiency towards improving system reliability and sustainability. Fact has been revealed by [5] who give a statement that the ultimate goal for a manufacturing system is guaranteeing an efficient production while providing functions needed by society in a sustainable and reliable way. It provides a new perspective

regards the future development of TEL system. However, challenges still remain which can be categorized as follows:

- **Heterogeneity:** The heterogeneity is defined as system entities have different types of data model, properties, operation mechanisms and even different hardware and operating system [6]. As for TEL management especially for asset management and supervision, different data resources, operational information, past experiences and knowledges are characterized as heterogeneous resource and thus impose difficulties in integration.
- **Interoperability:** When it comes to interoperability, three perspectives can be identified [7], (1) organizational level: generic approaches and shared understanding of concepts, process, beliefs and terms [8]. (2) system level: interconnection between independent systems. (3) data level: consider the data properties include data format, data availability, data representation and semantic meanings. With respect to asset supervision, the interoperability challenges are inevitably presented at all three levels.
- **Integrated decision making:** Logistics asset is considered as large-scale and complex equipment. If a malfunction of single component or process has not been detected and corrected timely, it could lead to an expensive downtime and furthermore impose a great impact on the entire logistic activities. Consequently, a system with decision support becomes an essential element to provide relevant users a consistent understanding regards the system status and enabling an effective planning and execution of maintenance, such functionality could be referred as integrated decision making.

To cope with above mentioned challenges, in this paper, a context-aware supervision system is proposed which is used for information integration and supervision of large-scale asset service in TEL domain. The key ICT enablers are semantic web and autonomous agent. The ontology is used to model the semantic connections for heterogeneous data and various entities in supervision domain, thus enable information integration, data filtering and problem decomposition for specific supervision tasks. The usage of agent system intends to provide intelligent diagnosis and decision making functionalities through agent intelligence and cooperation. The integration of ontology-MAS delivers a context-aware intelligent system

and its practical usage will be considered with a case study of intelligent belt conveying system supervision.

The remaining part of the paper is organized as follows: Section II will introduce the concept of context-aware supervision system (CASS) and the motivations behind it. Section III will provide key technological enablers that could support the implementation of a CASS system. Section IV will first present the system design from an abstract structure perspective and the design of each functional block is discussed. Section V will presents a case study of applying CASS for intelligent supervising of a large-scale belt conveying system. The conclusion and future work will be addressed in section VI.

II. CONTEXT-AWARE SUPERVISION SYSTEM

A widely recognized definition of context is given by [9] as *Context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves.* And a system can be termed as context-aware if *it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task* [9]. A context-aware system (CAS) adapts and provides relevant information and the most appropriate service to users in an active and autonomous manner while requires little interactions [10].

In this paper, we focused on investigating the potential of applying CAS for asset supervision service. In essence, the supervision service includes system monitoring, failure/abnormality diagnosis/prognosis and maintenance planning. Apart of being context-aware of delivering meaningful information to user, it also requires a transparent flow from data to supervision method. To achieve that, the objective of a supervision system is to deliver accurate and timely information regards system conditions and propose effective maintenance actions to ensure reliability and availability of the system. Its success relies on integrating different diagnosis/prognosis methods. However, such methods often have certain scope of applicability and input context. As such, it put additional requirements on CAS to systematically integrate and manage system supervision processes. To integrate CAS for system supervision and fulfill additional requirements, a novel concept of *Context-aware supervision* is defined:

A context-aware supervision system (CASS) should include a series of functionalities include monitoring, supporting and advising in relation to system events. It not only focuses on diagnosing and prognosis failures, but also responsible for managing and organizing system knowledge, reasoning facts, integrating resources and analyzing problems. As such, the failure context can be given in a more meaningful manner that delivers information include: the specification of fault condition, recorded data linked to it, maintenance or operating actions linked to it, users that responsible for it and method that been used to determine it.

The characteristics of *context-awareness* are presented at two levels in CASS: (1) the supervision method should be aware of the context information it operates upon; (2) the end

user should comprehend the created supervision context. In literature, several works have been established which attempt to consolidate the concept of context-aware with asset management such as context-awareness predictive maintenance [11], context-aware e-maintenance [12] and context-aware condition monitoring [10]. Two limitations are drawn from previous works: (1) the scope of applicability of proposed concept is limited given the fact that it only concerns partial aspect of the supervision process, For instance, the work of [11] concerns predictive maintenance, it lacks details regards how context been modeled and processed. (2) most works stay on a conceptual level which lacks sufficient technical details on how to put the concept into practice. Our work contributes to the literature by first introducing the concept of CASS, then we will discuss the technology been selected for putting such concept into action. Finally, a case study would demonstrate how the system works.

III. KEY TECHNOLOGICAL ENABLER FOR CASS

A. Context-modeling methodology

Intuitively, large amounts of context information are either acquired or derived from sensor devices. Normally, there exist gaps between raw data and the level of information which is useful to applications [13]. The context-modeling is used to bridge this gap by processing and transforming raw data before passed to context-aware services. Krummenacher et.al [14] have proposed several criteria for context model selection which include applicability, comparability, traceability, quality and so on. Meanwhile Hoareau et.al [13] conducted an extensive review for existing modeling choices such as key value models, makeup scheme, logic based models, object oriented models, ontology and so on. According to their in-depth discussion, the use of ontology is proposed to be the most expressive models to fulfill our requirements [15]. A formal definition of ontology is given by [16] as *an explicit specification of a conceptualization* which was used to describe a specific domain knowledge where concepts and relationships are unambiguously defined and checked. Recent works extensively applied ontology to facilitate the context modeling, its applicability covers domain include risk management of cold chain logistics [17], enterprise application [18], process supervision [19] and so on.

B. System supervision

A system supervision process considers providing users with decision support before/during/after the occurrence of system failure or abnormal situations. A typical supervision process consists of data acquisition, condition diagnosing/prognosis and maintenance planning. In this paper, rather than considering specific method or algorithm, we focus on how to provide a generic and adaptive environment to incorporate and integrate different methodologies and mechanisms operate together with flexibility and scalability.

Agent, as a tool in artificial intelligence domain, provides a way of dealing with complex engineering problem and

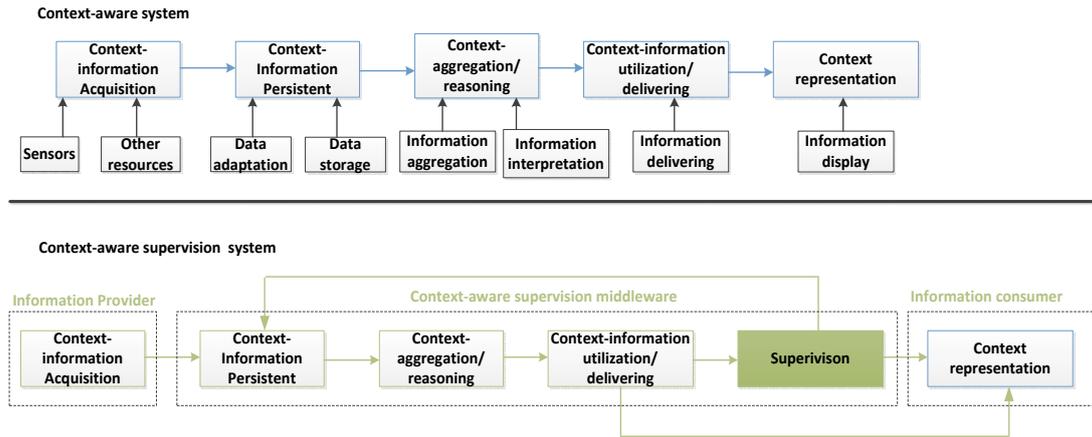


Fig. 1. System abstract architecture: a comparison between CAS and CASS

establishing adaptive system for decision making and information management through agent intelligence and collaborations [20]. State-of-arts demonstrate that agent system are largely applied to support system supervision functions, which include condition monitoring [21], risk management [22], e-maintenance [23] and so on. As such, agent technology is chosen as the key enabler for supervision system design, reasons are given as: (1) agent could cooperate and deploy on top of existing software. (2) In a multi-agent-system, agents could collaborate with each other to communicate and exchange information. (3) agent system could be deployed in distributed environment where new agent could easily join the system or leave the system as needed.

C. Ontology-agent integration

Key technological enablers have been chosen in previous section. We select ontology as the context-modeling method and agent system as the environment to support system supervision integration. In order to implement CASS, a next step is to consider the integration issues. In literature, several works have been established that concerns the integration of ontology and agent system. Dibley et.al [24] presented a work of building monitoring system where three ontologies are developed to capture the major semantics of a building environment and agent system is deployed to facilitate the monitoring tasks. For most of existing works, attentions are paid on using ontology to assist agent communication and knowledge retrieving. To implement CASS, potentials include information analysis, problem decomposition, agent status control are needed. To achieve this, a novel ontology-agent integrated framework is proposed, it will be elaborated in next section.

IV. SYSTEM FRAMEWORK

Fig 1 presents a comparison between a context-aware system and the proposed context-aware supervision system from an abstract structure perspective. A classical context-aware system follows five key processes [10]: (1) context information acquisition: gather information from virtual resources and

physical sensors; (2) context-information persistent: data filtering and storing ;(3) context-aggregation/reasoning: interpretation and transfer low-order data to high-level applicable information via aggregation and reasoning; (4) context information utilization/delivering: apply context information to implement application-specific service; (5) context representation.

All of the functional blocks from CAS are inherited and implemented in CASS. The major difference is distinguished from two aspects. **The supervision block:** The key objective of CASS is to assist system supervision with aspects include equipment monitoring, condition diagnosis and maintenance planning. Such tasks are unable to perform well by only using context-modeling(e.g. ontology reasoning). In most cases, it requires advanced platform/engine for decision making. As such, the supervision block is introduced. **The information flow:** The information flow is also adjusted. In CAS, the information flow follows an open loop style where data is gathered from ground layer, processed through each functional block and becomes context-aware. For CASS, a partial closed-loop is formed. In essence, the aggregated and processed context information will be the input source for supervision module. The output of supervision module will be feedback for further processing. It will be first stored in data base and then processed by context model. By doing so, not only the measured data from ground layer would be context-aware but also the supervised result will be aggregated with other relevant information together to make result meaningful to end user. Moreover, it would be helpful to use the returned supervision information to infer new knowledge and propose further actions.

A. Context Model Design

We design an ontology termed *ontoSupervision* to capture major concepts and relationships in the domain of system supervision. The schematic of *ontoSupervision* is given in Fig 2 and explanation of each taxonomy is given below:

(1) *System taxonomy* is the core concept that presents a description of the system. It includes notions of system

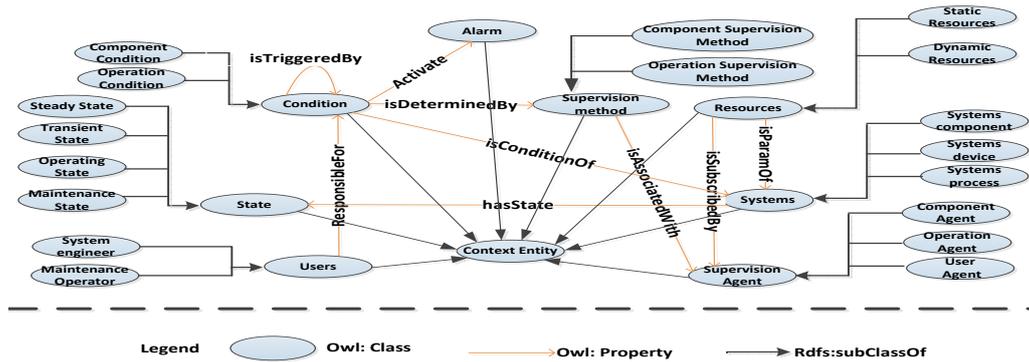


Fig. 2. Upper ontology taxonomy: definition of classes and object properties

fundamental components and operations that needs to be supervised. In addition, the subclass *system devices* contains peripheral devices. Other taxonomies either directly or indirectly connect with system through well defined relationships. (2) *Condition taxonomy* incorporates the notion of condition in the system. Two subclasses are included, namely operation condition and component condition. The former one concerns the system abnormal condition during operations and the latter one addresses the physical condition of system at different levels (component, instrument and equipment). (3) *Resources taxonomy* represents all relevant information resources that need to be accessed by supervision method. It is composed of two subclasses, namely static resources and dynamic resources. The former one can be thought as the resource that does not change over time such as system specification, historical information and system configurations. The latter one represents the notion of resources that change in real time, such as data acquired from sensor devices and any updated supervision results. (4) *Supervision agent & method* represents all available agents been deployed in current system and its associated supervision method. In essence, it serves as a bridge that connects context model with agent system. (5) *State taxonomy* represents possible state of the system. In current model, four states are considered namely maintenance state, transient state and operation state and steady state. (6) *Alarm taxonomy* represents the notion of possible alarm level that been activated by supervised conditions in the system. (7) *User* represents the information consumers in the system. It specifies the responsibilities and point of interests of respective user.

B. Agent system design

Regardless of the models, scopes and design tools, all supervision methods require system measurements as input and generate results which transform system conditions (operation and component condition) as supervision result. Such common structure allows the supervision methods to be represented as supervision agent. The multi agent system can be perceived as a wrapper which provides environment for different supervision intelligence to perform supervision tasks. It also enable integrated decision making by taking the advantages of agent

communication and collaboration. In the proposed framework, three kinds of agent are developed:

(1) **Supervision Agent:** Two categories of agent groups are considered as supervision agent. The first one is termed healthiness agent (HA) which is responsible for fault diagnosis at different level of system granularity. Single HA could be used to assess the condition of piece of equipment while multi HAs could work together for evaluating the overall healthiness of the whole system by consolidating different conditions. Another one is termed operation agent (OA) which is used to capture the abnormality during system operation. Typically, it is used to identify the abnormal deviation from normal operations or improper configurations. The agent intelligence, scope of interest, input information and responsibility are determined by its associated methods. (2) **Information Mediator Agent:** The information mediator is used to manage and control the agent execution and interactions. Its necessities are given as: It serves as an information portal for supervision agents; It keeps an active connection between agent system and ontology knowledge model. (3) **User Agent:** It contains the information consumer of the system. Any on-going supervision conditions will be relayed to it via IMA. Sophisticated GUI will connect with it to provide end user a friendly interface.

C. Agent-Ontology integration

The key of agent-ontology integration is achieved via the interaction between information mediator agent and ontology knowledge base. Such interactions aim at manipulating ontology to acquire information and knowledges where actions include create, read, update and delete entities in ontology. We identified three major processes:

- *Information acquisition:* In this case, ontology is treated as a hybrid database which is used to locate and retrieve information. A typical scenario can be that when a new sensor measurement is available in ontology, the IMA could retrieve it by executing well defined query template. An example of a query template is shown in Fig 3.
- *Knowledge acquisition:* It fully utilizes the reasoning capability of an ontology model. When certain information is available, the ontology could infer new knowledge

```

SELECT ?par ?agent ?system ?method ?timeStamp
?state ?measurement
WHERE {
    ?par onto:isSubscribedBy ?agent.
    ?par onto:isParametersOf ?system.
    ?agent rdf:type onto:AgentName.
    ?method onto:isAssociatedWith ?agent.
    ?system onto:hasState ?state.
    ?par onto:hasTimeStamp ?timeStamp.
    ?par onto:isMeasuredBy ?measurement.}

```

Fig. 3. Example of information acquisition template

by executing context-rules. For instance, if a belt idler temperature is over 70 degree, the ontology could use rules to determine that such context indicates a idler is in fault condition.

- *Knowledge reasoning and agent control*: As discussed previously, a partial closed loop is formed in the structure of CASS. The key motivation is given that any returned supervised information could be further processed by ontology. And the inferred knowledge could be useful in coordinating agent activities. For instance, a misalignment condition often occurs during the running of a belt conveyor and such condition could be induced by multiple reasons(improper power supply, overloading and so on) and such casualty relationships could be pre-defined in the ontology via proper object properties. By doing so, when a misalignment condition is supervised and returned, the ontology could running context rules to find the relevant condition relate with it. Consequently, the associated agent will be activated to allow a depth investigation of the root cause.

V. IMPLEMENT CASS FOR LARGE SCALE MATERIAL HANDLING SYSTEM

A. case demonstration

Belt conveying system is widely accepted as a major equipment in continues material handling domain. Its usages are well developed in various logistics domains, such as container/dry bulk terminals, airport and mine industry. Normally, the BCS is deployed in an open and harsh environment, as such, major components could suffer severe damage as system ages. Consequently, a monitoring and supervision system with decision support is essential to help users(from operations, maintenance, reliability and other departments) gain a consistent understanding about the system status and enabling effective planning and execution of maintenance. Due to the limit space of the paper, we demonstrate a typical fault supervision process- belt tear condition supervision which is made up for 85% among all system component damages for a BCS [25].

B. Scenarios

This scenario demonstrates the CASS capability of the system. Specifically, when inspection of tear shape is available, the system should analyze the damage level and propagation pace of the damage by intelligent supervision method, and create decisions in the form of possible maintenance activities

and/or warning/alarm message if needed. We identify three key processes of implementing the context-aware supervision service for belt tear condition:

- *Context modeling*: It concerns extending the upper ontology (ontoSupervision) with definition of new entities for application purpose. Such extension is termed domain-specific ontology and partial illustration for BCS supervision (ontoBeltCon) is depicted in Fig 4. The semantic meaning is given as: a tear shape (TS) *isMeasuredBy* a human inspection tool(HIT), which *isParameterOf* belt (belt section01). To supervise the tear condition, a belt tear supervision agent (BTSA) is designed. The BTSA *hasAssociatedMethod* belt tear supervision method (BTSM1). For supervision purpose, TS and a belt tear condition log (BTCL) *isSubscribedBy* BTSA. Upon successful decision making, a belt tear condition (BTC) is supervised which *isDeterminedBy* BTSM1 and *activate* alarm (alarm level 1). Finally the BTC *isResponsibleFor* user (maintenance operator 01).
- *Context supervision*: After context information are collected and pre-processed by ontology, the agent intelligence should be invoked. For a belt tear condition supervision, a fuzzy logic based approach is applied [25]. Decisions are made based on the current tear shape measurement and history inspection log for the same shape. Two indicators (belt wear index and inspection frequency index) are provided to deliver a consistent understanding and interpretation of the supervision result and give straight forward suggestions for possible maintenance actions.
- *Response actions*: The supervised condition will be send back to ontology model for further processing before finally delivered to end users. In essence, it will use the supervision indicators to quantify the alarm level by running defined rules. For the given scenario, the rules can be given as:

$$BeltTearCondition(?condition), greaterThan(?level, 0), hasWearIndex(?condition, ?level), lessThanOrEqual(?level, 0.7) \rightarrow AntiHealthCondition(?condition)$$

VI. CONCLUSION AND FUTURE WORK

In this paper, a novel concept of *context-aware supervision* and its associated implementation techniques are proposed. The motivation behind the concept is to enable an efficient and transparent information flow for asset supervision tasks. We implement such system for supervision of a large-scale material handling system to demonstrate its major functionalities and potential usage in the domain of logistics. Future works include further extending the ontology model to incorporate more generic entities and concept in the system supervision domain. Moreover, to cope with more complex diagnosis/prognosis problem and enable more sophisticated decision making engine, the agent intelligence should be future investigated.

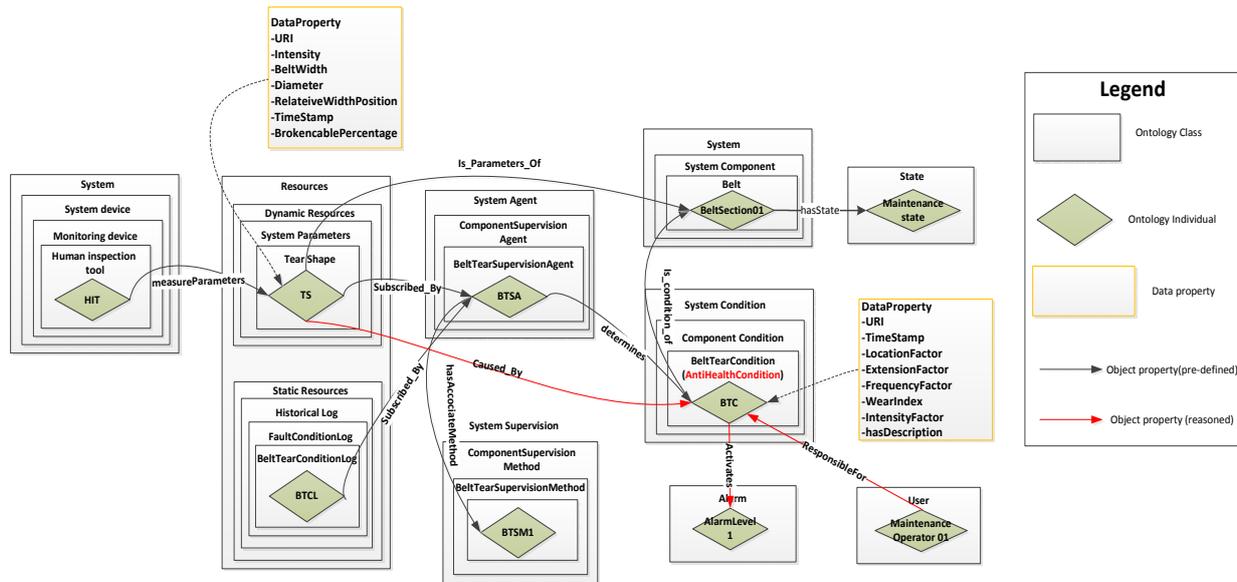


Fig. 4. OntoBeltCon configuration for belt tear condition supervision: include class, subclasses, data properties and object properties

REFERENCES

- [1] H. N. Chiu, "The integrated logistics management system: a framework and case study," *International Journal of Physical Distribution & Logistics Management*, vol. 25, no. 6, pp. 4–22, 1995. doi: 10.1108/09600039510093249
- [2] A. Gunasekaran, E. W. T. Ngai, and T. C. E. Cheng, "Developing an e-logistics system: a case study," *International Journal of Logistics Research and Applications: A Leading Journal of Supply Chain Management*, vol. 10, no. 4, pp. 333–349, 2007. doi: 10.1080/13675560701195307
- [3] U. Arnold, J. OberlÄnder, and B. Schwarzbach, "Advancements in cloud computing for logistics," in *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*, M. P. M. Ganzha, L. Maciaszek, Ed., 2013, pp. pages 1055–1062.
- [4] D. La Cruz, A. López, H. Vecke, and G. Lodewijks, "Prognostics in the control of logistics systems," in *IEEE International Conference on Service Operations and Logistics, and Informatics, 2006. SOLI'06*. IEEE, 2006, pp. 1–5.
- [5] S. Takata, F. Kirmura, F. van Houten, E. Westkamper, M. Shpitalni, and etc, "Maintenance : changing role in life cycle management," vol. 1, no. 1, 2004.
- [6] G. Thomas, G. R. Thompson, C.-W. Chung, and et.al, "Heterogeneous Distributed Database Systems for Production Use," *ACM Computing Surveys (CSUR) - Special issue on heterogeneous databases*, vol. 22, no. 3, pp. 237–266, 1990.
- [7] L. S. Winters, M. M. Gorman, and A. Tolk, "Next generation data interoperability: It's all about the metadata," in *IEEE Fall Simulation Interoperability Workshop*, 2006.
- [8] T. Clark and R. Jones, "Organisational interoperability maturity model for c2," in *Proceedings of the 1999 Command and Control Research and Technology Symposium*, 1999.
- [9] A. K. Dey, "Understanding and using context," *Personal and ubiquitous computing*, vol. 5, no. 1, pp. 4–7, 2001.
- [10] D. Galar, A. Thaduri, M. Catelani, and L. Ciani, "Context awareness for maintenance decision making: A diagnosis and prognosis approach," *Measurement*, vol. 67, pp. 137–150, 2015.
- [11] B. Schmidt, D. Galar, and L. Wang, "Current Trends in Reliability, Availability, Maintainability and Safety," 2016.
- [12] P. Pistofigdis and C. Emmanouilidis, "Profiling context awareness in mobile and cloud based engineering asset management," in *Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services*. Springer, 2012, pp. 17–24.
- [13] C. Hoareau and I. Satoh, "Modeling and processing information for context-aware computing: A survey," *New Generation Computing*, vol. 27, no. 3, pp. 177–196, 2009.
- [14] R. Krummenacher and T. Strang, "Ontology-Based Context Modeling," *leice Transactions On Information And Systems*, vol. E90-D, no. 8, pp. 1262–1270, 2007.
- [15] R. Schmohl, U. Baumgarten, and D.-G. M, "A Generalized Context-aware Architecture in Heterogeneous Mobile Computing Environments A Generic Context-aware Architecture," *Wireless and Mobile Communications*, pp. 118–124, 2008.
- [16] N. Guarino, D. Oberle, and S. Staab, "What is an ontology?" in *Handbook on ontologies*. Springer, 2009, pp. 1–17.
- [17] K. Kim, H. Kim, S.-K. Kim, and J.-Y. Jung, "i-RM: An intelligent risk management framework for context-aware ubiquitous cold chain logistics," *Expert Systems with Applications*, vol. 46, pp. 463–473, 2015.
- [18] D. Nadoveza and D. Kiritisis, "Ontology-based approach for context modeling in enterprise applications," *Computers in Industry*, vol. 65, no. 9, pp. 1218–1231, 2014.
- [19] S. Natarajan and R. Srinivasan, "Implementation of multi agents based system for process supervision in large-scale chemical plants," *Computers and Chemical Engineering*, vol. 60, pp. 182–196, 2014.
- [20] G. Kovács and K. Grzybowska, "Supply chain coordination between autonomous agents: A game-theory approach," in *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 5. IEEE, 2015, pp. 1623–1630.
- [21] I. Mahdavi, B. Shirazi, N. Ghorbani, and N. Sahebjamnia, "IMACS: Design and implementation of an intelligent multi-agent system for monitoring and controlling quality of cement production processes," *Computers in Industry*, vol. 64, no. 3, pp. 290–298, 2013.
- [22] R. J. Dawson, R. Peppe, and M. Wang, "An agent-based model for risk-based flood incident management," *Natural Hazards*, vol. 59, no. 1, pp. 167–189, 2011.
- [23] R. Yu, B. lung, and H. Panetto, "A multi-agents based e-maintenance system with case-based reasoning decision support," *Engineering applications of artificial intelligence*, vol. 16, no. 4, pp. 321–333, 2003.
- [24] M. Dibley, H. Li, Y. Rezgui, and J. Miles, "An ontology framework for intelligent sensor-based building monitoring," *Automation in Construction*, vol. 28, pp. 1–14, 2012.
- [25] G. Lodewijks and J. Ottjes, "Application of Fuzzy Logic in belt conveyor monitoring and control," *International Materials Handling Conference (Beltcon) 13*, pp. 1–13, 2005.