# The Ontologically based Model for the Integration of the IoT and Cloud ERP Services

Darko Andročec, Ruben Picek and Marko Mijač
*Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, Varaždin, Croatia*

Keywords: ERP, Cloud ERP, Internet of Things, Iot, Integration, Interoperability, Semantic Web.

Abstract: On-premise and Cloud ERP systems have become a backbone of almost all businesses. Another recent trend currently in focus of both industry and academy is Internet of Things. The integration of Cloud ERP and the Internet of Things (IoT) should be looked as a new shift in business effectiveness and will have a great momentum in future. In this work, we propose the ontologically based model for the integration of the IoT and Cloud ERP systems by using Semantic Web services. To semantically annotate things as services, we plan to use recently published W3C's SSN and SOSA ontologies. Furthermore, we plan to extend mentioned ontologies to include classification and descriptions of Cloud ERP APIs. Our integration model proposes usage of Semantic web services and AI planning technique to semi-automatically compose IoT and Cloud ERP services.

## 1 INTRODUCTION

ERP systems have, from their very start, aimed at being the IT backbone for business processes in enterprises. To achieve that, they had to employ state of the art information technology, and keep an eye on the future trends and developments in industry. One of the best examples of that is how ERP solutions were heavily influenced by recent advances in cloud computing technology. These advances resulted in emergence of Cloud ERP systems, which implied reshaping of technological, business and other aspects of ERP systems. The *software-as-a-service* (SaaS) model introduced changes from technological point of view, making the ERP systems more flexible, scalable and available from anywhere. It also changed our view and use of ERP systems, previously as a product, and nowadays as a service. This allowed the shift to subscription business model, which eliminated the need for up-front capital investments, making the ERP solutions more accessible to small and medium enterprises.

Another recent trend that is currently being in focus of both industry and academy is *Internet of Things* (IoT). The term originated in 1999 from proposal of uniquely identifiable interoperable connected objects with radio-frequency (RFID) technology (Ashton, 2010). Of course, over time, the concept included more and more evolving technologies. Today, when we speak about IoT, we speak about billions of "*things*" connected to a vast network, which collect data by sensing their physical environment, share this data with interested parties, and intervene into concrete situations. Possibilities of Internet of Things are so vast and diverse, that it is hard to foresee all possible applications of the technology. However, some notable examples include smart homes, smart cities, transportation, healthcare, agriculture, enterprises etc.

While it is still relatively novel concept for most enterprises, IoT has a potential to again reshape ERP systems, by making Cloud ERP more flexible and intelligent. According to research by IDC (Rian van Heur, 2015), 40% of data by 2020 will be machine-generated, with 20 to 50 billion of connected devices fuelling that growth. This will make Cloud ERP systems more complex, but it will also enable a unique point for adding value business.

The core characteristics of IoT and Cloud ERP complement each other. On one hand IoT provides interfaces to physical environment in which the enterprise operates, thus being able to collect vast amount of data. On the other hand Cloud ERP ensures vast resources to storage, analyse and process this data. IoT can provide Cloud ERP with real-time data about the state of the performed business processes and involved resources (people, equipment, tools,



Andročec, D., Picek, R. and Mijač, M.
The Ontologically based Model for the Integration of the IoT and Cloud ERP Services.
DOI: 10.5220/0006763104810488
In *Proceedings of the 8th International Conference on Cloud Computing and Services Science (CLOSER 2018)*, pages 481-488
ISBN: 978-989-758-295-0

materials and products) in the real world enterprise setting. Cloud ERP can use this data to help people respond in a timely manner to possible malfunctions, inefficiencies, safety and security risks, and other issues at the operational level. Cloud ERP can also use this data to support management activities, by providing advanced analysis, statistics, visualization, past trends, and predictions. These applications of IoT technology in enterprises can be categorized as follows (Lee and Lee, 2015): (1) monitoring and control, (2) Big Data and business analytics, and (3) information sharing and collaboration.

In order to achieve synergy between Cloud ERP and IoT, there has to be a way of integrating these two technologies. In this paper, we propose the model of Cloud ERP and IoT integration, based on semantic web services and AI planning technique for composition. The rest of the paper proceeds as follows. In section 2 the related work about integration of Cloud ERP and IoT technology is listed. Section 3 contains description of methodology for model development. Description of the model itself can be found in section 4. In the last section we discuss the proposed model, and provide our conclusions.

## 2 RELATED WORK

Cloud, ERP systems and IoT technologies are each separate fields with large body of existing research. However, their integration has a great potential in providing benefits for each technology. Pairing of Cloud computing and ERP systems has already proved itself as a great move, which reshaped the whole ERP market. Today, all major ERP vendors, both large and small, offer their solutions in a form of *Software-as-a-Service* (SaaS), i.e. Cloud ERP solutions, which have numerous advantages over traditional on-premises solutions. These advantages include (Johansson et al., 2015): lower upfront costs, lower TCO, availability, flexibility, integration with other services, etc.

Integration of IoT into Cloud ERP systems looks similarly promising. One of the main reasons for that is the complementarity of these technologies. Authors (Botta et al., 2014) investigated integration of Cloud computing and IoT, and consider following characteristics as complementary:

- **Storage Resources** – IoT produces a large amount of non-structured or semi-structured data. Cloud, on the other hand offers almost unlimited capacity for storing that data. In big data terminology we can say that IoT represent big data source and Cloud represents platform for managing big data.

- **Computational Resources**– IoT devices have no or very limited computational capabilities. This is why collected data is transferred to Cloud which has has required resources to process this data.

- **Communication resources** – Cloud has built-in real-time solutions for connecting, tracking, monitoring and controlling practically anything from anywhere.

Above stated complementary characteristics are perfectly valid also for Cloud ERP and IoT. If anything, ERP components augments this complementarity with being one of the software systems most dependent on large amount of business data. This can be nicely seen in following definition of IoT in enterprise context (Haller et al., 2009): "*A world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in business processes. Services are available to interact with these 'smart objects' over the Internet, query their state and any information associated with them, taking into account security and privacy issues*". Cloud acts as intermediate layer between the "*things*" and the ERP system, where it hides all the complexity and the functionalities necessary to implement latter.

According to (Boza et al., 2015), interoperability of Cloud ERP and IoT can be seen as interaction between ERP system and other, internal or external systems. Same authors proposed two perspectives of interoperability, the first one considering *technological* aspects such as web services, SOA, Cloud computing, IoT etc., and the second one considering *business* aspects such as BPM, BPR; virtual enterprises, references models etc. In our paper, we focus on *technological* perspective. The difficulties of legacy systems to exchange information with each other within the company have been overcome by the implementation of ERP systems (Boza et al., 2015).

While the need for integration between Cloud ERP with IoT and other systems in general is apparent, interoperability remains a significant issue. For example, issues with *compatibility and integration with other existing systems* is recognized as one of the major barriers in adopting Cloud ERP systems (Picek et al., 2017). Different approaches for mitigating that problem have been proposed in literature. Most of them are based on Semantic Web, for example, SOCRADES (de Souza et al., 2008) is a middleware for business integration, focused on integrating web service enabled devices with ERP systems and other enterprise applications.

Architecture for effective integration of the Internet of Things in enterprise services has been proposed by (Spiess et al., 2009). Meyer et al. (2013) identify and integrate IoT devices as a type of resources in business processes. Song et al. (2010) propose application layer solution as a semantic middleware for interoperability between IoT devices. Alexakos et al., (2016) present an approach to integration between IoT and manufacturing processes based on semantics. Zhuming Bi et al. (2014) investigate the impact of IoT to modern manufacturing in Enterprise Systems. Molano et al. (2017) proposed a meta-model for integration of IoT, Social networks, Cloud and Industry 4.0. The novelty of our approach is usage of existing cloud ERP application programming interfaces (APIs) and IoT services that can be semantically annotated and semi-automatically translated into AI planning method returning plan how to compose the mentioned two types of services. The main aim of our proposal is to enable service-level interoperability among IoT services and cloud ERP APIs.

# 3 METHOD

Semantic Web is often used in research papers and research projects to tackle interoperability problems among different systems, models, and frameworks, e.g. integration of cloud computing services (Androcec and Vrcek, 2016a) or integration of IoT services (Androcec and Vrcek, 2016b). For this reason and our prior works, we have also chosen Semantic Web as a main method in our proposal of the model for the integration of the Cloud ERP and IoT services. The main idea of the Semantic Web is to provide coherent data model that is a part of the web infrastructure (Berners-Lee et al., 2001). One data item can point to another using standard links. The fundamental concepts of Semantic Web are (Berners-Lee et al., 2001): the AAA slogan (anyone can say anything about any topic), open world (it is assumed that there is always more information than known), and non-unique naming (the same entity can have more names).

Semantic Web consists of a number of modelling languages that are organized in layers. The basis of Semantic Web is the Resource Description Framework (RDF) used for representing information about resources that can be identified by URIs (W3C, 2004). However, we have chosen the more expressive Web Ontology Language (OWL 2), because it is designed to represent rich and complex knowledge, and is most often used in related interoperability/integration papers. The main elements of Web Ontology Language (OWL 2) are classes, properties, individuals, and data values (W3C, 2009). The most important tools when working with OWL are ontology editors (we have used the open-source tool Protégé) used to create and edit ontologies, and reasoners (we have used reasoner embedded to the Protégé tool) to infer logical consequences.

OWL is mostly used to define ontologies that describe a certain domain. The ontologies are often used to tackle interoperability problems (Uschold and Gruninger, 1996). The most cited definition of ontology is: "An ontology is an explicit specification of a conceptualization"(Gruber, 1993). The ontology defines basic concepts and their relationships in a specified domain of interest. Noy and McGuinnes define ontology as "formal explicit description of concepts in a domain of discourse" (Noy and McGuinness, 2001), together with their properties and restrictions. The ontologies are most often developed to share common understanding, reuse, separate, and analyse the existing domain knowledge, and make domain assumptions explicit (Noy and McGuinness, 2001). In the next sub-section we will briefly describe the Semantic Sensor Network Ontology (Compton et al., 2012) , that is mostly used in the literature as a basis for IoT ontology development.

## 3.1 SSN and SOSA Ontology

In October 2017, W3C published the new version of their Semantic Sensor Network Ontology (W3C, 2017) that will be used as a basis for our ontology for annotation of Cloud ERP and IoT services. "The Semantic Sensor Network (SSN) ontology is an ontology for describing sensors and their observations, the involved procedures, the studied features of interest, the samples used to do so, and the observed properties, as well as actuators. SSN follows a horizontal and vertical modularization architecture by including a lightweight but self-contained core ontology called SOSA (Sensor, Observation, Sample, and Actuator) for its elementary classes and properties" (W3C, 2017).

SOSA extends the original scope of the SSN ontology to include classes and properties for actuators and sampling (see Figure 1.). Given the increased interest to use Semantic Web technology on individual things (sensors, actuators, and platforms), SOSA is lightweight and does not use the more complex language elements of the SSN (W3C, 2017). SOSA aims at broadening the target audience (web developers) and application areas that can make use of Semantic Web ontologies (W3C, 2017). The new SSN introduces additional classes and relations on top of SOSA to model the capabilities of sensors and actuators and the compositionality of systems (W3C, 2017).

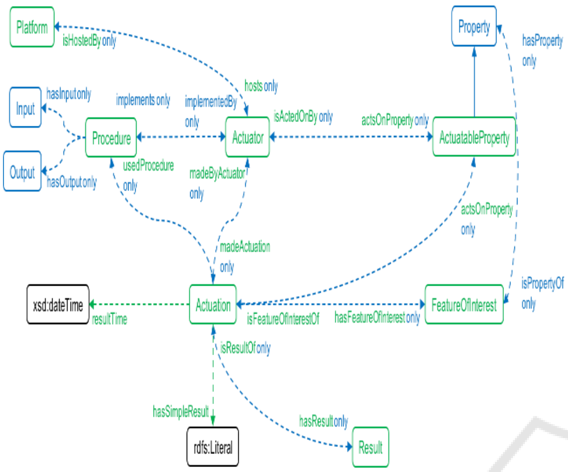


Figure 1: Actuator perspective of the SOSA (W3C, 2017).

## 3.2 Semantic Web Services

All of the main Cloud ERP vendors expose some of the services of their solutions as application programming interfaces (APIs) in form of the SOAP or RESTfull web services. For example, Microsoft Dynamics NAV provides SOAP and OData web services. An example that lists the operations of SOAP web service to work with customer object is depicted at the Figure 2. Also, the functionalities of the sensors and actuators are mostly expressed in the form of the web services, either individually (per Web thing) or through IoT middleware or brokers, e.g. Global Sensor Network (Aberer et al., 2006).

Current web services provide only syntactical descriptions, so web service integration must be done manually. Semantic web services are the integration of Semantic Web and service-oriented architecture implemented in the form of web services. Semantic web services are aimed at an automated solution to the following problems: description, publishing, discovery, mediation, monitoring and composition of services. To implement Semantic Web service, new languages are used: OWL-S (Semantic Markup for Web Services), Service Modeling Ontology (WSMO), or lightweight approaches such as WSMO-Lite, SAWSDL, MicroWSMO, hRESTS, and SA-REST.

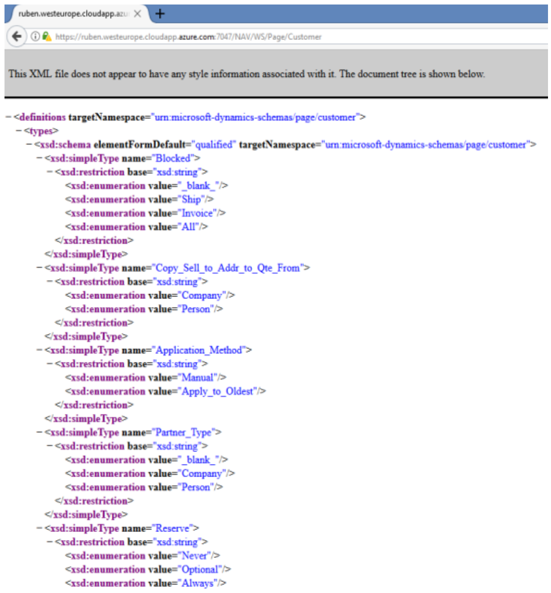


Figure 2: Sample of the MS NAV 2016 SOAP API.

## 3.3 AI Planning Method

AI planning is one of the most promising approaches to solve a problem of automated Semantic Web service composition. Sirin et al. proved the semantic correspondences between the SHOP2 planner and OWL-S, and they showed how one can use SHOP2 planner to compose web services (Sirin et al., 2004, p. 2). Hierarchical Task Network (HTN planning) is the AI planning technique that is most widely used for practical applications (Goyal, 2010). For this reason, we have used the HTN planning in our model to compose ERP and IoT services. RESTfull and SOAP web services can be translated into planning axioms that can be used to semi-automatically compose services relevant to stated problem or desired steps defined in the planning problem file. The implementation details are shown in the Section 4.1.

# 4 MODEL FOR THE IoT AND CLOUD ERP SERVICES INTEGRATION

Semantic Web is the dominant method and technique to integrate different systems, so we have chosen it in our work to propose model of integration of Cloud ERP APIs and things as a service. To compose the semantically annotated web services, we have chosen AI planning method.

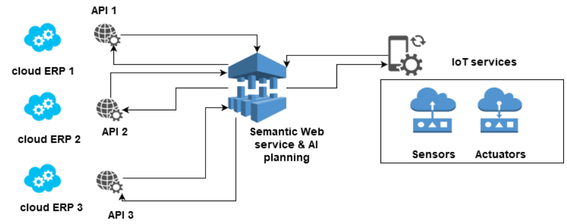


Figure 3: IoT and cloud ERP integration model.

## 4.1 Integration Model

Our model is described in Figure 3, and the methods and tools for each layer are described in the Table 1. The main aim of our model is to enable integration and interoperation of IoT services and application programming interfaces (APIs) defined by Cloud ERP providers. Things (sensors, actuators and complex things) and their functionalities are exposed as IoT services (SOAP or RESTful services) individually or through IoT (often service oriented and cloud based) middleware such as Global Sensor Network (Aberer et al., 2006), openIoT (Soldatos et al., 2015), Hydra (Eisenhauer et al., 2009), Xively etc. All main Cloud ERP providers offer APIs in form of SOAP or RESTful services which enable integration of third-party application, systems or data with Cloud ERPs. In our model, services are semantically annotated using Semantic Web services standard. After that, Semantic Web services can be semi automatically composed. For this purpose, we use AI planning technique. The similar approach was used for similar purposes in the existing literature: for example, to integrate cloud services of different cloud providers (Androcec et al., 2015), and to enable interoperability of different IoT services (Androcec and Vrcek, 2016b). The main advantage of our proposed model is that it enables integration of the IoT services with the chosen Cloud ERP solution or multiple Cloud ERP solutions.

To semantically annotate web services, SAWSDL will be used in this work. It enables the usage of the semantic annotation by specifying references to semantic models such as SSN and SOSA ontologies mentioned before in this work. The concept from the semantic models can be referenced from WSDL or XML schema. A model reference can be used with every WSDL element, but its meaning is defined in SAWSDL only for interface, operation, fault, xs:element, xs:complexType, xs:simpleType and xs:attribute (W3C, 2007). The same annotation on a WSDL operation or fault gives semantic information about the annotated operation or fault, and it provides a classification of the interface on a WSDL interface.

The support for data mediation in SAWSDL is provided by using the 'liftingSchemaMapping' and 'loweringSchemaMapping' attributes on web service message input and output elements to create mappings with the ontology concept with which input or output is associated with (Nagarajan et al., 2007).

Table 1: Methods and techniques of the proposed model.

| Layer | Proposed methods, tools or techniques |
|---|---|
| Physical (sensors, actuators, and complex things) | Native interfaces of things (serial ports, WiFi, cloud, etc.). Optional usage of IoT middleware (e.g. GSN, openIoT). |
| Web service layer (IoT services and APIs of Cloud ERP providers) | SOAP and RESTfull services |
| Semantic Web service layer | SAWSDL, XSLT |
| Semi-automatic composition of Cloud ERP APIs and IoT services | AI planning technique using JSHOP2 tool |

Web operations and their inputs/outputs will be semantically annotated, and SAWSDL and XSLT will be used to define service type mappings, similar to the work (Androcec et al., 2015) where semantic annotation were used to annotate APIs of different cloud providers. Data mediation will be ontology based. We use the new version of the mentioned W3C's ontologies: SSN and SOSA to annotate thing as service. We also plan to upgrade the mentioned ontologies to include Cloud ERP APIs functionalities, inputs and outputs to enable interoperability between Cloud ERP APIs and semantically annotated things as a service.

SAWSDL provides its lifting and lowering schema mapping features to map XML elements to the ontology and back. Use of cross-Cloud ERP and IoT services concepts for data types in the ontology simplifies mappings, and enables the creation of new mappings and possible transformations, when new Cloud ERP offer or new IoT service is used, or when specific API is changed. This is a more flexible approach than direct mapping and transformation approach used in web service composition languages like BPEL. The most critical part of this approach is the requirement for user/administrator to create valid and meaningful mappings and transformations.

To compose the semantically annotated web services, we have chosen the AI planning technique. Concretely, we have used JSHOP2 tool (Ilghami, 2006, p. 2). JSHOP2 is a Java version of Simple Hierarchical Ordered Planner (SHOP). It is used to generate sequential plans. It is based on ordered task decomposition where tasks are planned in the same order as later in execution (Ilghami, 2006, p. 2). The objective of JSHOP2 and other HTN planners is to accomplish a set of tasks where each task can be decomposed, until primitive tasks (Ilghami and Nau, 2003) are reached. The inputs of JSHOP2 are a planning domain and a planning problem. In JSHOP2, primitive tasks are called operators whose name must begin with an exclamation mark. The body of an operator consists of precondition (must be satisfied to execute the action), delete list (set of properties that will be removed), and add list (set of properties that will be added) (Ilghami, 2006, p. 2). Solving a planning problem in JSHOP2 is done in three steps: the domain description file is compiled into Java code, the problem descriptions are converted into Java class, and the second Java class should be executed to initiate the planning process and inspect the planning results.

## 5 DISCUSSION AND CONCLUSIONS

The possible application of the proposed model for IoT services and Cloud ERP APIs can be done by choosing a business processes in one ERP module (e.g. maintaining and services) and connecting resources (e.g. equipment) with IoT devices (e.g., temperature or movement sensors) that will collect real-time data. Based on these data, through ERP system we can try to accelerate and optimize everyday activities and proactively increase business effectiveness and efficiency. This will be added functionality of ERP system achieved through custom forms (e.g. page in Microsoft NAV). Business rules will be triggered on some values and workflows. Using tools for business intelligence, we can analyse the various performance indicators, create business reports or new segments of monitoring for selected business processes through the ERP system. For example, we can use IoT service that returns temperature from the sensor attached to a specific machine in a specific production hall. If company uses e.g. Microsoft Dynamics NAV 2016 ERP system, we can find in its documentation that it provides SOAP web operation *void Create(ref Entity entity)* that creates a single record. We can semantically annotate the mentioned two services and use our proposed method to store temperature data in ERP database that can be further used for some analysis or for a new action request in the ERP system.

IoT can bring positive impact on the companies' performance by increasing operational efficiency and by reducing operating costs. The connected things allow cost reduction, e.g. if extraordinary maintenance (detection of abnormal parameters by integrated sensors) and malfunctions of the machines are reported immediately and integrated with used Cloud ERP solutions. The IoT is also able to improve the inventory management.

The main contribution of our work is a cloud-IoT integration state-of-the-art and the proposal of the model for the integration of IoT services and Cloud ERP APIs at the service level. Our model uses Semantic Web technologies, ontologies (SSN, SOSA, and the Cloud ERP API ontology), SAWSDL to define Semantic Web services, and AI planning technique to semi automatically compose defined IoT services and Cloud ERP APIs. Many ERP vendors provide a way to integrate IoT data with their systems, but big disadvantage is that they provide proprietary tools and methods applicable only for their solution. What if the customer needs or wants to switch Cloud ERP solution? Our approach is more general and flexible because it does not rely on proprietary technology or specific Cloud ERP vendors.

As a future work, we plan to implement the proposed model and develop proof-of-concept software to integrate and use sensors data in different Cloud ERPs. For this purpose, we plan to design various related experiments. We also plan to develop the ontology for classification and descriptions of Cloud ERP APIs. Integration of IoT and business process management suite's (BPMS) services seems an interesting future research subject, as sensors or/and actuators can accept roles in the workflow. IoT covers a huge range of devices which produce useful information for organizations, so we believe that integration of IoT services and Cloud ERP systems is important research and professional topic.

## REFERENCES

Aberer, K., Hauswirth, M., Salehi, A., 2006. The Global Sensor Networks middleware for efficient and flexible deployment and interconnection of sensor networks (Technical Report No. LSIR-REPOR T-2006-006). Ecole Polytechnique Federale de Lausanne, Lausanne.

Alexakos, C., Anagnostopoulos, C., Kalogeras, A.P., 2016. Integrating IoT to manufacturing processes utilizing semantics. IEEE, pp. 154–159. https://doi.org/10.1109/INDIN.2016.7819150.

Androcec, D., Vrcek, N., 2016a. Ontologies for Platform as Service APIs Interoperability. Cybern. Inf. Technol. 16. https://doi.org/10.1515/cait-2016-0065.

Androcec, D., Vrcek, N., 2016b. Thing as a Service Interoperability: Review and Framework Proposal. In *IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*. IEEE, pp. 309–316. https://doi.org/10.1109/FiCloud.2016.51.

Androcec, D., Vrcek, N., Kungas, P., 2015. Service-Level Interoperability Issues of Platform as a Service, In *IEEE World Congress on Services (SERVICES 2015)*. IEEE, New York City, pp. 349–356. https://doi.org/10.1109/SERVICES.2015.60.

Ashton, K., 2010. The "Internet of Things" Thing. RFID J.

Berners-Lee, T., Hendler, J., Lassila, O., 2001. The Semantic Web. Sci. Am. 284, 29–37.

Boza, A., Cuenca, L., Poler, R., Michaelides, Z., 2015. The interoperability force in the ERP field. Enterp. Inf. Syst. 9, 257–278. https://doi.org/10.1080/17517575.2013.866697.

Botta, A., de Donato, W., Persico, V., Pescapé, A., 2014. On the Integration of Cloud Computing and Internet of Things, In *2014 International Conference on Future Internet of Things and Cloud*, Barcelona, pp. 23-30.

Compton, M., Barnaghi, P., Bermudez, L., García-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., Huang, V., Janowicz, K., Kelsey, W.D., Le Phuoc, D., Lefort, L., Leggieri, M., Neuhaus, H., Nikolov, A., Page, K., Passant, A., Sheth, A., Taylor, K., 2012. The SSN ontology of the W3C semantic sensor network incubator group. Web Semant. Sci. Serv. Agents World Wide Web 17, 25–32. https://doi.org/10.1016/j.websem.2012.05.003.

de Souza, L.M.S., Spiess, P., Guinard, D., Köhler, M., Karnouskos, S., Savio, D., 2008. SOCRADES: A Web Service Based Shop Floor Integration Infrastructure. In: Floerkemeier, C., Langheinrich, M., Fleisch, E., Mattern, F., Sarma, S.E. (Eds.), The Internet of Things. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 50–67. https://doi.org/10.1007/978-3-540-78731-0_4.

Eisenhauer, M., Rosengren, P., Antolin, P., 2009. A Development Platform for Integrating Wireless Devices and Sensors into Ambient Intelligence Systems. In: SECON Workshops '09. 6th Annual IEEE Communications So*ciety Conference on Sensor, Mesh and Ad Hoc Communications and Networks Workshops*. IEEE, Rome, Italy, pp. 1–3. https://doi.org/10.1109/SAHCNW.2009.5172913.

Goyal, A., 2010. Real-Time Planning Using HTN in Stealth Game.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowl. Acquis. 5, 199–220. https://doi.org/10.1006/knac.1993.1008.

Haller, S., Karnouskos, S., Schroth, C., 2009. The Internet of Things in an Enterprise Context. In: Domingue, J., Fensel, D., Traverso, P. (Eds.), Future Internet – FIS 2008. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 14–28. https://doi.org/10.1007/978-3-642-00985-32.

Ilghami, O., 2006. Documentation for JSHOP2 [WWW Document]. URL http://sourceforge.net/projects/shop/files/JSHOP2/ (accessed 7.5.13).

Ilghami, O., Nau, D.S., 2003. A General Approach to Synthesize Problem-Specific Planners (Technical Report No. CS-TR-4597). University of Maryland, College Park, Maryland.

Johansson, B., Alajbegovic, A., Alexopoulo, V., Desalermos, A., 2015. Cloud ERP Adoption Opportunities and Concerns: The Role of Organizational Size. IEEE, pp. 4211–4219. https://doi.org/10.1109/HICSS.2015.504.

Lee, I., Lee, K., 2015. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. Bus. Horiz. 58, 431–440. https://doi.org/10.1016/j.bushor.2015.03.008.

Meyer, S., Ruppen, A., Magerkurth, C., 2013. Internet of Things-Aware Process Modeling: Integrating IoT Devices as Business Process Resources. In: Salinesi, C., Norrie, M.C., Pastor, Ó. (Eds.), Advanced Information Systems Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 84–98. https://doi.org/10.1007/978-3-642-38709-8_6.

Molano, J.I.R., Lovelle, J.M.C., Montenegro, C.E., Granados, J.J.R., Crespo, R.G., 2017. Metamodel for integration of Internet of Things, Social Networks, the Cloud and Industry 4.0. J. Ambient Intell. Humaniz. Comput. https://doi.org/10.1007/s12652-017-0469-5.

Nagarajan, M., Verma, K., Sheth, A.P., Miller, J.A., 2007. Ontology Driven Data Mediation in Web Services. Int. J. Web Serv. Res. 4, 104–126. https://doi.org/10.4018/jwsr.2007100105.

Noy, N.F., McGuinness, D.L., 2001. Ontology Development 101: A Guide to Creating Your First Ontology.

Picek, R., Mijač, M., Andročec, D., 2017. Acceptance of Cloud ERP Systems in Croatian Companies: Analysis of Key Drivers and Barriers. In *20th International Scientific Conference on Economic and Social Development (Book of Proceedings)*.

Rian van Heur, 2015. Internet of things could make ERP flexible, intelligent and real-time. Computerweekly.

Sirin, E., Parsia, B., Wu, D., Hendler, J., Nau, D., 2004. HTN planning for Web Service composition using SHOP2. Web Semant. Sci. Serv. Agents World Wide Web 1, 377–396. https://doi.org/10.1016/j.websem.2004.06.005.

Soldatos, J., Kefalakis, N., Hauswirth, M., Serrano, M., Calbimonte, J.-P., Riahi, M., Aberer, K., Jayaraman, P.P., Zaslavsky, A., Žarko, I.P., Skorin-Kapov, L., Herzog, R., 2015. OpenIoT: Open Source Internet-of-Things in the Cloud. In: Podnar Žarko, I., Pripužić, K., Serrano, M. (Eds.), Interoperability and Open-Source Solutions for the Internet of Things. Springer International Publishing, Cham, pp. 13–25.

Song, Z., Cardenas, A.A., Masuoka, R., 2010. Semantic middleware for the Internet of Things. IEEE, pp. 1–8. https://doi.org/10.1109/IOT.2010.5678448.

Spiess, P., Karnouskos, S., Guinard, D., Savio, D., Baecker, O., Souza, L.M.S. de, Trifa, V., 2009. SOA-Based Integration of the Internet of Things in

Enterprise Services. IEEE, pp. 968–975. https://doi.org/10.1109/ICWS.2009.98.
Uschold, M., Gruninger, M., 1996. Ontologies: principles, methods and applications. Knowl. Eng. Rev. 11, 93. https://doi.org/10.1017/S0269888900007797.
W3C, 2017. Semantic Sensor Network Ontology (W3C Recommendation No. OGC 16-079).
W3C, 2009. OWL 2 Web Ontology Language Primer (W3C Recommendation).
W3C, 2007. Semantic Annotations for WSDL and XML Schema (W3C Recommendation). W3C.
W3C, 2004. RDF Primer (W3C Recommendation).
Zhuming Bi, Li Da Xu, Chengen Wang, 2014. Internet of Things for Enterprise Systems of Modern Manufacturing. IEEE Trans. Ind. Inform. 10, 1537–1546. https://doi.org/10.1109/TII.2014.2300338.