Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 60 (2017) 86 – 91

www.elsevier.com/locate/procedia

27th CIRP Design 2017

Integrated data model and structure for the asset administration shell in Industrie 4.0

Erdal Tantik*, Reiner Anderl *

Department of Computer Integrated Design, Technische Universität Darmstadt, 64287 Darmstadt, Germany

* Corresponding author. Tel.: +49 6151 16-21844; fax: +49 6151 16-21793. *E-mail address:* tantik@dik.tu-darmstadt.de

Abstract

The increasing demand for highly customized products in connection with shortened product life cycles requires the manufacturing industry to be eminently flexible, whereas low production costs are crucial to persist in the competition of the global markets. To meet this requirements, cyber-physical systems (CPS) are applied into the production process, aiming for interconnected and self-managing smart factories, which can incorporate external and internal conditions for the autonomous adaptation to gain optimized results. This is achieved through a bi-directional information flow between all important components such as machines, products, control programs and off-site assets. Therefore it is essential to standardize communication interfaces and enhance interoperability between CPS of all variations. This paper presents an approach to combine the specification of the World Wide Web Consortium (W3C) with the guidelines of the Plattform Industrie 4.0 (I4.0), thus obtaining a uniform structure for industrial CPS. Based on the recommended asset administration shell for I4.0-components, the required functionality is identified and allocated to different segments. The five main segments include the functionality for security, representation, communication to external CPS, communication to internal assets and a section for additional applications to enhance the capabilities. By using a standardized protocol for the configuration and representation based on the object memory model of the W3C, a significant interoperability between I4.0-components and conventional Internet-of-Things can be realized. The proposed structure is applied in a use case to simulate the adaptation and remote maintenance of a production robot.

Peer-review under responsibility of the scientific committee of the 27th CIRP Design Conference

Keywords: Cyber-Physical Systems; Industrial Internet; Industrie 4.0; Internet of Things; Remote Maintenance

1. Introduction

A significant research subject of the past century was the development of new methods to manage increased lot-sizes in the production process. This subject pervaded through the decades, changing the production structure from former hand-crafted products to fully automated production lines, thus increasing the affordability and expanding the customer circle. The developed methods furthermore had to consider the continuously decreasing product life cycles (PLC), which require an appropriate degree of flexibility to adapt to recent developments and integrate new technologies into the production [1,2]. Thereby flexibility was primarily determined by the time-to-volume, which measures the necessary time and efforts for the alteration of a mass production system to integrate new technologies. In recent years the customer demands have increasingly shifted from mass products to personalized devices. Automotive manufacturers for instance have to offer a huge amount of specifications and optional features which can be chosen and changed by the customer. This leads to a tremendous expansion of the product variety. In combination with the lean-production methods designed to avoid inventory, the result is a significant reduction of lot-sizes. The final stage of this process is a production system which is continuously adapting to every new order, while the value stream is constantly changing. To remain a high productivity, the adaptation itself has to be automated, thus individual orders can be processed seamless within the production system. The

Peer-review under responsibility of the scientific committee of the 27th CIRP Design Conference
doi:10.1016/j.procir.2017.01.048

opposing requirements of highly automatized yet flexible systems can be accommodated with the application of advanced information and communication technologies into the production process. The application is enabled by the remarkable developments on the field of computer science. Overall the hardware components are getting smaller, cheaper and more efficient. Combined with the perpetually expanding Internet availability and improving web technologies, an extensive communication network has been created, thereby supporting the fundamental requirements for huge data collection. The impact is most apparent in the private sector and customer goods. Advanced computer technologies are already used on the shop floor, mostly integrated into the production stations as embedded systems, which can control machines and processes on a restricted level [3]. Additionally the technologies are indispensable at the office and administration level. The management of the complex product creation process extensively uses advanced information and communication technologies, mainly for Product Data Management and Enterprise-Resource-Planning. Nonetheless the processes are controlled in the hierarchical structure of the automation pyramid. Consequently the next step is to improve the production machines and the supporting components into autonomous systems, which comprise external and internal conditions for the adaption process. The adaptation process may require complex software programs. Therefore all relevant components of the production system have to be transformed into cyber-physical systems (CPS). A CPS can identify itself, set up a connection with surrounding network systems, interact with other CPS and have additional functionality. In the context of the production process each batch, production machine and separate asset can be CPS. For instance the workstations can identify the products to obtain specific information about each order. In combination with the information about the surrounding CPS the whole production process can be adapted and optimized to enhance the value stream. Accordingly the rigid structure of the automation pyramid is transformed into a more flexible decentralized control pattern (Fig. 1).

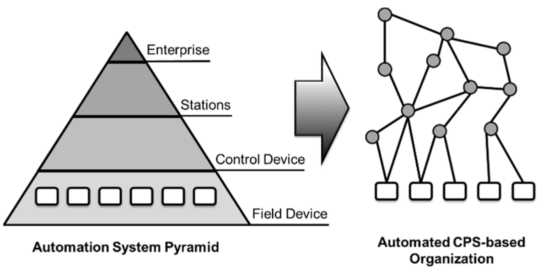

Fig. 1. The usage of cyber-physical systems enables to transform the automation pyramid into a decentralized and interconnected control system [4].

Thereby the decentralized control system with autonomous CPS can adapt the processes to each individual order, thus enabling custom tailored products at low expenses. Furthermore the concept provides additional benefits and capabilities. All information about the production process can be stored and used for the optimization of subsequent processes [5,6]. Additionally the interconnectivity combined with the stored information can be used for the following steps of the product life cycle and enable new service-oriented business models [7].

2. State of the art

Because of the lack of consistent standards, currently there is a huge amount of devices and technologies labeled CPS, embedded systems (ES) and Internet of Things (IoT). The transition between the different categories is often seamless, thus the differentiation mostly depends on their purpose. The term CPS is widely used for industrial devices with the ability to access other CPS and process complex information, thus allowing for a less restricted and more autonomous adaptability than common control systems and ES [6]. In contrast the term IoT is generally used for industrial and private devices, which explicitly connect to the Internet and web services. Nonetheless it is beneficial to define the CPS as the general term, while considering IoT and ES as subgroups, thus emphasizing their specific capabilities [8]. The fundamental requirement for an efficient production process based on CPS is the interoperability, thus ensuring the bi-directional information flow by CPS of different manufacturers [9]. This presupposes the standardization of the communication interfaces and data formats. Because of the vast variety of CPS, standardizations are generally very difficult. To develop and support standardized interfaces, numerous association are working together e.g. the Advanced Manufacturing Partnership 2.0 of the American Government, Plattform Industrie 4.0 of the German government and the Industrial Internet Consortium (IIC) supported by numerous industrial enterprises [7,8]. Furthermore the World Wide Web Consortium (W3C) increasingly supplies standards for IoT. The main focus of the IIC consortium lies on the general industrial usage of the internet, including in transportation, energy, healthcare and manufacturing. Industrie 4.0 on the other hand predominantly focuses at the manufacturing processes and closely related areas [10]. For the definition of compulsory standards the Plattform Industrie 4.0 specifies a new industrial subgroup of CPS [11].

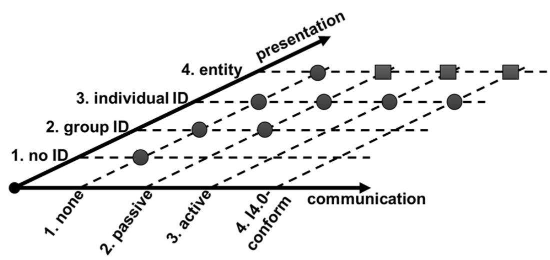

Fig. 2. The categorization of CPS (round dots) and I4.0-components (square dots) according to the Plattform Industrie 4.0 [11].

To define which category of CPS fulfills the requirements, two characteristics must be considered. The first characteristic describes the ability of the CPS to communicate with its environment whereas the second describes how it is presented and whether it can be identified by other CPS. While the communication has no influence on the presentation, a minimum degree of identifiability is necessary for certain communication with other CPS. The resulting categories of CPS can be seen in Fig. 2.

Each CPS can be classified in a CP-category. For instance CP23 is an asset which may have an affixed visual code. The visual code can be an individual ID, referencing additional information and functionality on a server system. The new subgroup defined for industrial standardization is named Industrie 4.0-components (I4.0-components). The square dots in Fig. 2 show which characteristics currents systems require to be I4.0-components [11,12]. Every I4.0-component has to represent an entity. The entity can be physical or digital asset. The definition makes it possible to classify assets without integrated and accessible computers as I4.0-component, if the digital representation of the asset is accessible on a separate computer. Thereby current production machines can be improved to I4.0-components with very limited expenses.

2.1. Framework of the Asset Administration Shell

The main feature of these I4.0-components is an Asset Administration Shell (AAS), which administrates the communication of the entity to other systems and contains the functionality. It has one internal, manufacturer-dependent interface to the entity and one standardized interface for the external communication. Therefore the concept of the AAS is highly suitable for standardization without interfering with the functionality of the entity. The administrated entity can consist of a group of devices or nested I4.0-components. The Plattform Industrie 4.0 defines further requirements for the AAS relating to services, security, quality and communication [13,12]. The ability to administrate nested I4.0-components can highly increase the complexity of the top-tier AAS, but allows for a very adaptable network of the entire production system. A partially accessible digital representation of all nested entities is compulsory. A comprehensive framework for the AAS is proposed by the Plattform Industrie 4.0, which will work as fundament for future standardizations [14]. The AAS consists of a body and a header (Fig. 3). The most important part of the header is the manifest. It contains administrative information for the communication with other CPS, the identification, the functionality and an index of all part-models and sub-components. Thereby the manifest is the fundamental information point for other I4.0-components to gain access to the AAS, thus providing the interoperability. The Body of the AAS is separated into different part-models. A part-model can be a data model, a group of function or a component. Each part-model has a header of its own, which is part of the overall body manifest. The headers have to be standardized, referencing the specifics and capabilities. The body of the part-model contains the actual payload, which carries the information and conducts the methods. Another important part of the body is the component manager, which is an interface to access the payload of the modules. Apart from accessing the payloads, the component manager may have additional capabilities e.g. to maintain the AAS.

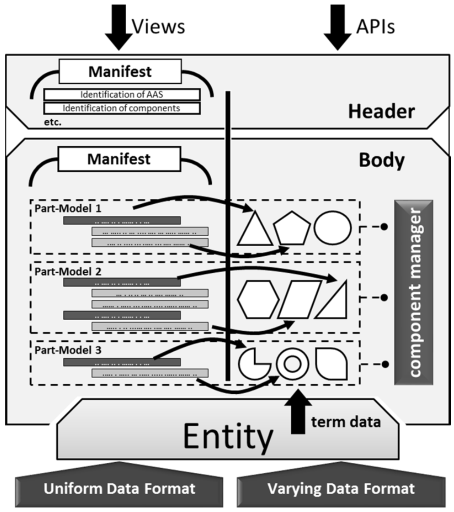

Fig. 3. Proposed framework of the Administration Shell by Plattform Industrie 4.0 [14].

3. Structure of the Asset Administration Shell

Even though the definitions of the Plattform Industrie 4.0 shows the framework of the AAS, the technical structure is not defined thus allowing for an implementation based on device-specific requirements. However by applying the requirements of the Industrie 4.0 to the main assets in the production process, the AAS can be subdivided into five segments:

- External Interface
- Authentication and Security
- Data Management
- Functionality
- Administration
- Internal Interface

The external interface administrates the data flow with the local I4.0-components and other interconnected systems. Therefore the communication has to be highly standardized to assure interoperability. Currently service-oriented paradigms e.g. RESTful and SOAP, as used for common web-services, are proposed for the communication [14]. The external interface is connected to a segment for the authentication and security. Depending on the entity-specific requirements, a huge amount of existing methods for IT-security can be implemented in this segment to enable an appropriate degree of security. In accordance with the guidelines for I4.0-components, the AAS has to manage numerous part-models, each containing a header and a payload. Depending on whether the payload of the part-model is a data package or an executable application, two separated segments can be defined. The segment for the data management consists of several modules, each containing information about an aspect of the AAS. One module is mandatory and contains header information as described in Fig. 3. Because of its role as a manifest, the module must be updated automatically about the status of the AAS. The other modules within the data management segment can be subdivided into three categories: They either store data about one aspect of the current I4.0-component, provide additional data about an application within the functionality segment or they reference nested I4.0-components. In some use cases, there can be numerous tiers of nested I4.0-components. These components often have to contain data from every step of the product life cycle. The top-tier AAS has to be able to represent or reference each of the nested I4.0-components [14]. This can result in a huge amount of data, thus it can be beneficial to use a product data management software as the data management segment. An automated addition and subtraction of modules enables a continuously changing network of I4.0- components. A similar adaptability can be provided by the functionality segment, which is a platform for different application. By adding and updating additional applications, the capabilities of the AAS can be highly improved. A similar concept has already been developed and implemented, demonstrating how IoT-devices can access an app storage to download additional programs [15]. With a standardized functionality segment, the concept can be applied to I4.0-components. This enables the design of AAS-templates, which can be customized for all kinds of assets by adding the required applications and data modules. The segments are interconnected through a central administration, which manages the data flow. The internal interface manages the communication between entity and AAS. Considering the huge variations of CPS and the different integration methods of the AAS within the component, a standardization is not preferable. Thereby a high applicability of the AAS to different CPS can be retained. For standardized AAS-templates uniform interfaces may be necessary. The resulting structure can be seen in Figure 4.

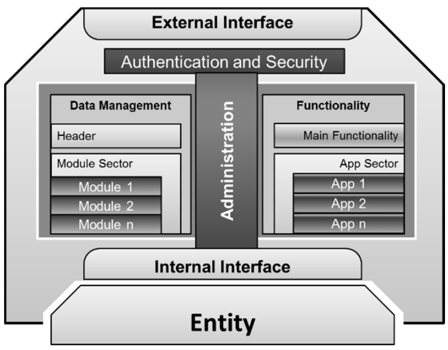

Fig. 4. Structure for the I4.0-component with separated data modules and functionality applications for a flexible adaptation.

One of the main characteristics of CPS and consequently of I4.0-components is the ability to store information of their product life cycles. Therefore it is beneficial to design the modules as digital object memories (DOMe). Considering the specific requirements of IoT-devices, W3C developed the object memory model (OMM), which is a block-based DOMe with highly standardized meta-information for the automated access [16]. The structure of the OMM can be seen in Fig. 5. Each OMM consists of one header for the identification of the memory, one table of contents (ToC) to quickly analyze the content of the memory, namely referencing the blocks, and an unlimited number of blocks containing the payload. A very useful feature of the Blocks is either to store the payload locally or reference its location on remote systems, thus allowing for a distributed storage of data. Thereby, a huge amount of information can be stored on external servers in order to keep the internal memory of devices small. Accordingly the structure of the OMM is very similar to the part-models of the Plattform Industrie 4.0.

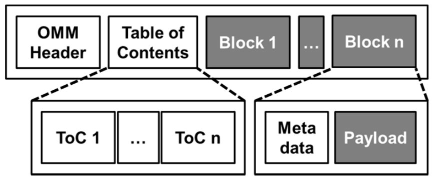

Fig. 5. Structure of the object memory model for the automated access of IoT [16].

Based on the integrated component data model [5] and the OMM, a new data model was developed specifically for I4.0-components. The resulting component data model (CDM) can be seen in Fig. 6.

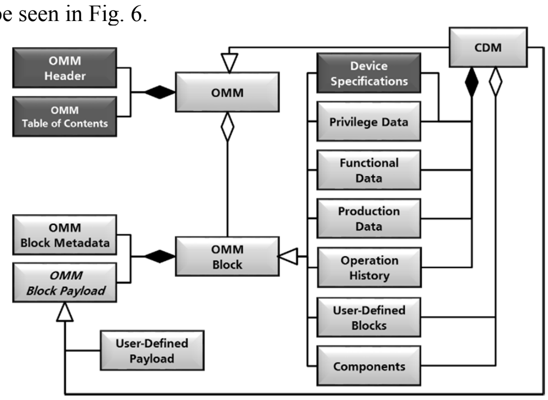

Fig. 6. Structure of the component data model as a subclass of the object memory model.

The CDM is a specialization of the OMM and consists of at least one header, one ToC and five specific blocks. The first block contains device specifications in accordance to the required manifest of I4.0-components. Thereby the header, the ToC and the device specifications conjointly form the header of the AAS. The privilege data block contains the information about groups, roles, views and privileges of connected systems to control the access to the I4.0-component. To enable an autonomous and self-managing network, each AAS has to provide information about its capabilities. The functional data block describes the core functionality and the applications of the AAS in a machine-readable format. Furthermore it may contain information about the invocation of the functionality, thus enabling remote maintenance and control. The functional data block can furthermore provide handbooks and additional tools to provide information for the user. The production data block contains information about each step of the production process. The subsequent usage data is stored in operation history block. The production data block combined with the operation history block are able to store the entire life cycle data of each individual component. Thereby the adaptability, functionality and applicability of the I4.0-component can be highly increased, allowing for numerous new services and business models such as remote maintenance and monitoring [6]. Furthermore user-defined blocks can be defined, thus enabling the adaptation of the I4.0-component. The CDM can represent data of nested I4.0-components. This is achieved by a separate component block, which contains a CDM in the payload. The nested CDM in the top-tier AAS can be a downsized copy of the CDM in the nested AAS. Because of the adaptability of the OMM, the header can be applied to the CDM without modifications.

4. Implementation

For the implementation a use-case with a central remote maintenance platform (CRMP) for the communication to I4.0-components was developed. Considering the capabilities of I4.0-components, the CRMP can be designed as an AAS. Thereby the same interfaces as used between different local I4.0-components can be used for the communication with the CRMP.

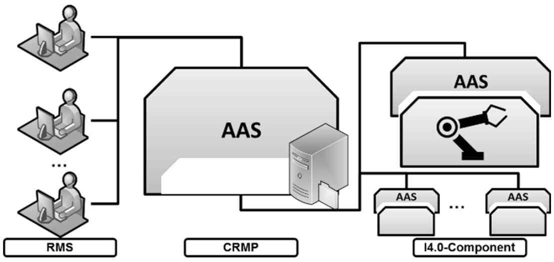

Fig. 7. Use case of a central remote maintenance platform for numerous I4.0-components.

The data management segments of the CRMP represent the different I4.0-components. A CDM has been developed, representing the CRMP on the top-tier. The CDM contains the digital representation of the nested I4.0-component. The functionality segment of the CRMP has one application for the administration for the remote maintenance. Thus the CRMP is not able to carry out simulations but can provide the data for the clients to do so on their systems. With this limitation, the CRMP is not able to run component specific external software, thereby reducing security risks and requirements for the implementation. If necessary, this limitation can be repealed by installing additional applications in the functionality segment. With this structure, the CRMP can be modified into a testing platform to carry out simulations of the I4.0-components. This design enables additional use-cases such as a data marketplace, a remote monitoring system and an external data storage for I4.0-components with the same central platform but different applications. Information can be transmitted to the CRMP regularly, and be accessed subsequently without interrupting the processes of the I4.0-component.

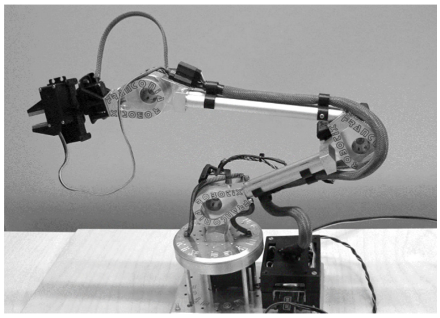

Fig. 8. For the Industrie 4.0 component a robot arm has been provided with an asset administration shell.

For the proof of concept, the proposed structure has been realized with two *Raspberry Pi 3* single-board computers. The general structure of both AAS is implemented in *Python 3*. Each segment is designed as an independent program to resemble proprietary software systems of different providers. For the communication between the segments the *ZeroMQ* library is used. For the external interface between the AAS the *Flask* web-framework is adapted to use RESTful-interface with HTTP. Thereby the interface can be used with a common browser e.g. on a mobile devices without additional applications. The maintained I4.0-component is a robot arm from *Franconia Robotix*. To reduce complexity of the communication, a new control software was developed, containing a user interface for the local control and an open port for the remote access. The robot arm and control software conjointly form the entity. The entity can progress independently from the AAS, which contains several applications to enhance the capabilities. Before a maintenance session starts, the AAS of the I4.0-component logs into the CRMP. The CRMP is monitored by various remote maintenance services (RMS) which are notified about their associated I4.0-components. The RMS can send a request for access to the CRMP. The requests are forwarded to the individual I4.0-components. The RMS and the I4.0-component have to agree on the conditions of access including the security standards. When the remote maintenance service is approved by both sides, a new thread is started, which manages the data progress for the session. The entire communication is realized as a client-server model with the CRMP as main server and I4.0-component and RMS as clients. The application thus forwards oncoming requests to the individual threads, which can progress the included information with a limited functionality. The threads have a limited access to the data management sector, thus can import specific parts of the CDM. The data models of each I4.0-component in the CRMP contain a user interface in HTML5. The user interface is custom-designed for the remote maintenance by the manufacturer of the I4.0-component. The RMS on the other hand is not required to install additional software. The access and maintenance of the I4.0-components is performed in the web-browser, very similar to accessing and using common web-site. Two methods have been developed for the remote maintenance. For the use case, the RMS is firstly provided with a visual impression of the I4.0-component. This can be done with a local smartphone camera next to the I4.0-component. A QR-code containing an URL is affixed to the robot. By scanning the QR-code, the smartphone requests a HTML5 document of the AAS, which accesses the camera and receives the video footage. The video stream is directed to the CRMP, from where the RMS can analyses the functionality of the robot. In the second step the RMS receives a HTML5 document very similar to the local graphical user interface of the I4.0-component. The HTLM5 document connects with the CRMP, thus enabling the remote control of the robot.

5. Conclusion and Outlook

In accordance to the guidelines for the Industrie 4.0 components a highly adaptable structure for the Asset Administration Shell (AAS) has been developed. The AAS provides independent segments for the data management and functionality, which are designed as platforms for additional applications to enhance the capabilities of the I4.0-component. Furthermore an integrated data model for the components has been developed. The design especially considers the requirements for automated access, the storage of information in all steps of the product life cycle, the inclusion of nested I4.0-components and the flexibility to customize the data model. The proposed structure was implemented for a robot arm and a for a service platform to remotely maintain the robot arm. Future research will focus on the standardization of communication between different I4.0-components to create self-managing systems, which can scan their environment for other I4.0-components and services. Thereby the I4.0-components will be enabled to manage the production process autonomously. Furthermore additional applications will be implemented into the AAS to increase their functionality.

Acknowledgements

This work has been funded by the German Federal Ministry of Education and Research (BMBF) in the project IUNO – National reference project on IT-Security in Industrie 4.0 (project number 16KIS0328).

References

[1] Shariatzadeh N, Sivard G, et al. Software Evaluation Criteria for Rapid Factory Layout Planning. Elsevier 2012. Procedia CIRP 3, p. 299 – 304
[2] Kia R, Khaksar-Haghani F, et al. Solving a multi-floor layout design model of a dynamic cellularmanufacturing system by an efficient genetic algorithm. Journal of Manufacturing Systems 33, Elsevier 2014; p. 218– 232
[3] Jóźwiak L. Embedded Computing Technology for Highly-demanding Cyber-physical Systems. International Federation of Automatic Control 2015; IFAC-PapersOnLine 48-4 (2015) 019–030
[4] Leitao P, Colombo A, et al. Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges. Elsevier 2015; Computers in Industry 81 (2016) 11–25
[5] Picard A, Anderl R. Integrated component data model for smart production planning. 19° Seminário Internacional de Alta Tecnologia 2014
[6] Mikusz M. Towards an Understanding of Cyber-Physical Systems as Industrial Software-Product-Service Systems. Elsevier 2014: Procedia CIRP 16 (2014) 385–389
[7] Anderl R. Industrie 4.0 – technological approaches, use cases, and implementation. De Gruyter Oldenbourg Automatisierungstechnik 2015; 63(10): 753–765
[8] Wang L, Törngren M, Onori M. Current status and advancement of cyber-physical systems in manufacturing. Journal of Manufacturing Systems, Elsevier 2015; Journal of Manufacturing Systems
[9] Gürdür H, El-Khoury J, et al. Making Interoperability Visible: Data Visualization of Cyber-Physical Systems Development Tool Chains. Elsevier 2016; Journal of Industrial Information Integration(2016), doi: 10.1016/j.jii.2016.09.002
[10] Soley R. Overview of the Industrial Internet Consortium. http://www.iiconsortium.org/berlin/Richard_Soley_Presentation.pdf, 2015
[11] Bangemann T, Bauer C, et al. Status Report Status Report Industrie 4.0 – Technical Assets Basic terminology concepts, life cycles and administration models. Düsseldorf, Frankfurt 2016; VDI – The Association of German Engineers, ZVEI – German Electrical and Electronic Manufacturers' Association
[12] Adolphs P, Epple U, et al. Status Report Reference Architecture Model Industrie 4.0 (RAMI4.0). Düsseldorf, Frankfurt 2015; VDI – The Association of German Engineers, ZVEI – German Electrical and Electronic Manufacturers' Association
[13] Feger U, Jänicke L, et al. Technical Overview, Secure cross-company communication. Federal Ministry for Economic Affairs and Energy (BMWi) 2016; Munich PRpetuum GmbH
[14] Adolphs P, Auer S, et al. Structure of the Administration Shell Continuation of the Development of the Reference Model for the Industrie 4.0 Component. Federal Ministry for Economic Affairs and Energy (BMWi) 2016; Spree Druck Berlin GmbH
[15] Kröner A, Hupert J, et al. Enabling Data Collections for Open-Loop Applications in the Internet of Things. International Journal on Advances in Internet Technology 2014; vol 7 no 1 & 2;
[16] Haupert J, Seißler M, et al. Object Memory Modeling. W3C Incubator Group Report 2011. https://www.w3.org/2005/Incubator/omm/XGR-omm-20111026/