

# Digital Enterprise Architecture with Micro-granular Systems and Services

Alfred Zimmermann<sup>1</sup>, Justus Bogner<sup>1,2</sup>, Rainer Schmidt<sup>3</sup>,  
Dierk Jügel<sup>1,4</sup>, Christian Schweda<sup>1</sup>, and Michael Möhring<sup>3</sup>

<sup>1</sup> Reutlingen University, Faculty of Informatics, Germany

<sup>2</sup> Hewlett Packard Enterprise, Boeblingen, Germany

<sup>3</sup> Munich University, Faculty of Computer Science and Mathematics, Germany

<sup>4</sup> University of Rostock, Faculty of Computer Science and Engineering, Germany

{alfred.zimmermann, dierk.jugel, christian.schweda}@reutlingen-university.de  
justus.bogner@hpe.com, rainer.schmidt@hm.edu, michael.moehring@hm.edu

**Abstract.** The digitization of our society changes the way we live, work, learn, communicate, and collaborate. This disruptive change interacts with all information processes and systems that are important business enablers for the context of digitization since years. Our aim is to support flexibility and agile transformations for both business domains and related information technology with more flexible enterprise information systems through adaptation and evolution of digital enterprise architectures. The present research paper investigates the continuous bottom-up integration of micro-granular architectures for a huge amount of dynamically growing systems and services, like Microservices and the Internet of Things, as part of a new digital enterprise architecture. To integrate micro-granular architecture models to living architectural model versions we are extending more traditional enterprise architecture reference models with state of art elements for agile architectural engineering to support the digitization of products, services, and processes.

**Keywords:** Microservices, Internet of Things, Digital Enterprise Architecture, Architectural Integration, Adaptable Architecture

## 1 Introduction

Digitization is the collaboration of human beings and autonomous objects beyond their local context using digital technologies. Digitization [16] further increases the importance of information, data, and knowledge as fundamental concepts of our everyday activities. By exchanging information human beings and intelligent objects are able to make decisions in a broader context and with higher quality. Social networks, smart portable devices, and intelligent cars represent only a few instances of a pervasive, information-driven vision [14] for the next wave of digital economy with digital products, services, and processes. Major trends for the digitization are investigated by [19] itemizing the digitization of products and services, context-sensitive value creation, consumerization of IT, digitization of work, and the digitization of business models. Microservices and the Internet of Things are

emerging to support next intelligent systems. They will shape future trends of business innovation and the next wave of information and communication technology. Biological metaphors of living and adaptable ecosystems [17] provide the logical foundation for self-optimizing and resilient run-time environments.

The technological and business architectural impact of digitization has multiple aspects, which directly affect adaptable digital enterprise architectures and their supported systems. Smart companies are extending their capabilities to continuously manage their changing business operating model by developing and maintaining Enterprise Architectures as the architectural part of a changing IT Governance [20]. Enterprise Architecture Management [7] with Services Computing [22] is the approach of choice to organize, build, utilize, and distribute capabilities for the digital enterprise architectures [23]. They provide flexibility and agility in business and IT systems. The development of such applications integrates Web and REST Services, Microservices, Internet of Things, Cloud Computing and Big Data management, among other frameworks and methods for semantic support.

Digitization of products and services requires the close alignment of business models and digital technologies for creative digital strategies and solutions, as well as for their digital transformation. Unfortunately, the current state of art and practice of enterprise architecture lacks an integral understanding and support of integrating a huge amount of micro-granular systems and services, like Microservices and Internet of Things, and the process of architectural adaptation for enterprise transformation. Our main motivation and the current presented work is to extend previous approaches of quiet static enterprise architecture to fit for flexible and adaptive digitization of new products and services and by introducing suitable mechanisms for collaborative architectural engineering and integration of micro-granular architectures. We report about our research to provide an adaptable digital enterprise architecture framework by continuously integrating relevant micro-granular information resources and their architectures for a fast growing number of digital products, services, and processes.

Our current paper is investigating the following research questions:

**RQ1:** What are architectural properties of Microservices and what are implication for integrating them into a digital enterprise architecture?

**RQ2:** How can we architect the Internet of Things and what is the resulting architectural composition context for the digitization of products and processes?

**RQ3:** How should the digital enterprise architecture be holistically tailored to integrate a huge amount of Microservices and Internet of Things architectures, considering the hypotheses that these micro-granular structures can be integrated into a consistent view of a digital enterprise architecture in a similar way?

The following Section 2 sets the fundamental context for digital enterprise architectures using Microservices. Section 3 focusses on architecting the Internet of Things for supporting the digital transformation. Section 4 presents with digital enterprise architecture our collaborative architectural reference and transformation approach and links it with specific architectural integration mechanisms for micro-granular systems and services. Finally, we summarize in Section 5 our research findings, our ongoing work in academic and practical environments and our future research plans.

## 2 Microservices Architecture

The term Microservices became popular in the last years and refers to a fine-grained style of service-oriented architecture (SOA) applications combined with several DevOps elements. James Lewis and Martin Fowler define a Microservice Architecture [8] as an approach for developing a single application from a suite of small services, each running in its own process and communicating with lightweight mechanisms, like HTTP. Microservices may additionally access NoSQL databases from on premise and optional Cloud environments.

These services are built around business capabilities and are independently deployable by an automated deployment pipeline. Typically, there is a bare minimum of centralized management of these services. Microservices may be written in different programming languages and can use different data storage technologies. As opposed to big monolithic applications, a single Microservice tries to represent a unit of functionality that is as small and coherent as possible. This unit of functionality or business capability is often referred to as a *bounded context*, a term that originates from Domain-Driven Design (DDD) [2].

However, Microservices also come with the need for a strong DevOps culture [1] to handle the increased distribution level and deployment frequency. Moreover, while each single Microservice may be of reasonably low complexity compared to a monolithic application, the overall complexity of the system has not been reduced at all. Gary Olliffe [9] distinguishes between the inner architecture and the outer architecture of Microservices (Fig. 1).

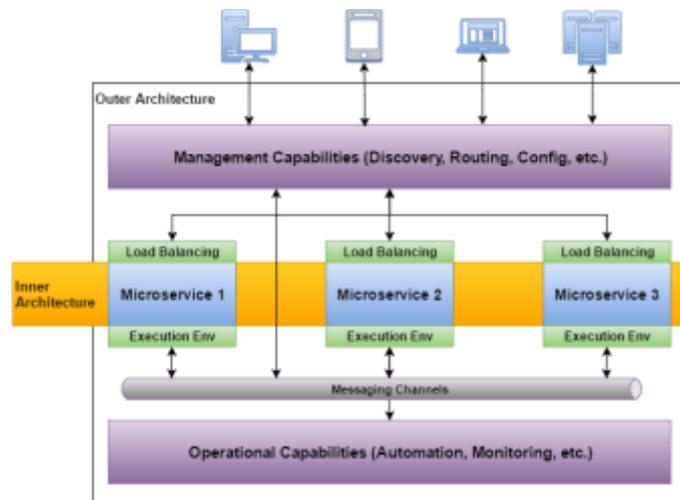


Fig. 1. Microservices Inner and Outer Architecture, based on [9]

By splitting up a big monolith into more fine-grained independent services, you shift most of the hindering complexity from the inner architecture to the outer architecture, where inter-service communication, service discovery, or operational capabilities have to be handled. The greatest benefits that come with Microservices

are the possibility to use the best-fitting technology for each *bounded context*. Typical examples are: increased application resilience (if one Microservice fails, the others may not be affected, at least if there is no chaining), independent and efficient scalability instead of replicating the complete monolith, and faster and easier deployment [1]. Especially the last advantage is an important step towards agility of business and IT systems.

Enabling technological heterogeneity is usually considered an advantage of Microservices [8] that allows the selection of the best tool for the job, reduces the possibility of lock-ins for outdated technology, and supports a culture of innovation and experimentation. However, Microservices also come with some risks for the organization. An explosion of technological diversity can quickly become overwhelming and unmanageable. Moreover, you are dependent on employees with the corresponding skills to handle these technologies and programming languages.

This is why most organizations that use Microservice Architecture either provide some very basic standardization without limiting their teams' choices too much or encourage the use of only a certain technology subset by offering comfortable tooling and infrastructure support for selected languages. Both approaches work reasonably well and prevent the existence of e.g. three different versions of Java or the use of six different web servers. This difficulty of keeping a healthy amount of governance and standardization while still allowing enough technological heterogeneity to not hinder innovation and agility can be addressed by Enterprise Architecture Management. However, classical approaches to enterprise architectures are often not flexible enough for the kind of diversity and distribution present in a Microservice Architecture.

### **3 Internet of Things Architecture**

The Internet of Things (IoT) fundamentally revolutionizes today's digital strategies with disruptive business operating models [19], and holistic governance models [20] for both business and IT. With the huge diversity of Internet of Things technologies and products organizations have to leverage and extend previous enterprise architecture efforts to enable business value by integrating the Internet of Things into existing business and computational environments. Reasons for strategic changes resulting from the Internet of Things [4] are:

Information of everything – enables information about what customers really demand,

- Shift from the thing to the composition – the power of the IoT results from the unique composition of things in an always-on,
- Always-connected information-rich environments,
- Convergence – integrates people, things, places, and information,
- Next-level business – the Internet of Things is changing existing business capabilities by providing a deeper way to interact, measure, operate and analyze business and IT.

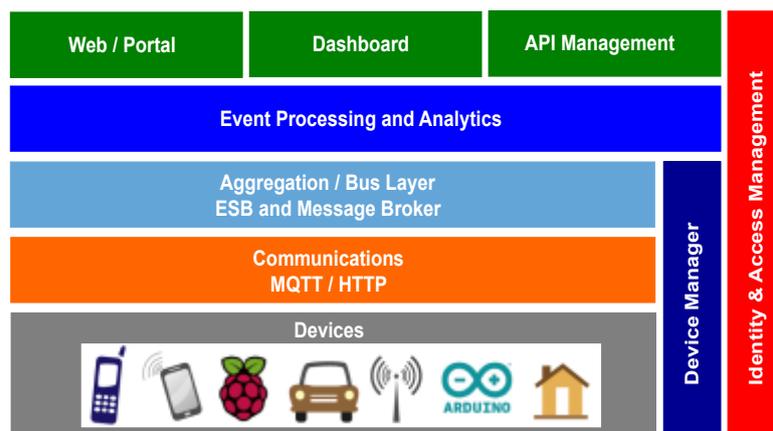
The Internet of Things enables a large number of physical devices to connect each other to perform wireless data communication and interaction, by using the Internet as

a global communication environment. The Internet of Things is the result of a convergence of visions [13]: Things-oriented vision, an Internet-oriented vision, and a Semantic-oriented vision. A cloud centric vision for architectural thinking of a ubiquitous sensing environment is provided by [4]. The typical configuration of the Internet of Things includes besides many communicating devices a cloud-based server architecture, which is required to interact and perform remote data management and calculations. In this way Internet of Things directly includes software and services into structures of digitized value chains.

Sensors, actuators, devices as well as humans and software agents interact and communicate data to implement specific tasks or more sophisticated business or technical processes [4], [14]. The Internet of Things maps and integrates real world objects into the virtual world and extends the interaction with mobility systems, collaboration support systems, and systems and services for big data and cloud environments. Furthermore, the Internet of Things fundamentally influences the Industry 4.0 [15] and adaptable digital enterprise architectures [23]. Therefore, smart products as well as their production is supported by the Internet of Things and can help enterprises to flexibly create customer-oriented products.

A main question is, how the Internet of Things architecture fits in a context of a service-based enterprise computing environment? A service-oriented integration approach for the Internet of Things is referenced in [23]. The core idea for millions of cooperating devices is, how they can be flexibly connected to form useful advanced collaborations within the business processes of an enterprise. The service-oriented architecture abstracts the heterogeneity of embedded systems, their hardware devices, software, data formats and communication protocols. A layered architecture structures the following bottom-up functionalities and prepares these layers for integration within an Internet of Things focused enterprise architecture: Devices Layer, Platform Abstraction Layer, Security Layer, Device Management Layer with Monitoring Services, Inventory Services, Service Lifecycle Management, Service Management Layer, and the Application Interface Layer.

A layered Reference Architecture for the Internet of Things is proposed in [21] and (Fig. 2). Layers can be instantiated by suitable technologies for the Internet of Things.



**Fig. 2.** Internet of Things Reference Architecture [21]

The Devices layer is the bottom layer, on which all other layers are built on. Devices are of different types, like cell and smart phones, cars, machines, house devices, and have to be connected directly or indirectly with the Internet. Each device needs an ID, which may be an UUID (unique identifier) provided by a device-chip, an UUID provided by the radio subsystem as a Bluetooth identifier, or a Wi-Fi MAC address, or an OAuth2 Token.

The Communications layer provides the devices' connectivity [13], [21], having to support typically multiple protocols for communication, like HTTP/HTTPS also supporting REST architectural styles, and lightweight protocols such as MQTT [<http://mqtt.org>], a publish-subscribe messaging protocol based on a broker model, and the Constrained Application Protocol (CoAP). CoAP enables IP and HTTP-based communications in a constrained environment. Mobility requirements and solutions for service-continuity in the Internet of Things in a mobile IPv6 environment are elaborated in [13], [21]. MQTT enables communication in lossy and intermittently connected networks on top of TCP. CoAP supports a RESTful application protocol over UDP with reduced footprint and is directly binary coded. Using the HTTP protocol for sending data to the device would cause an inefficient HTTP polling. Replacing it with the WebSocket protocol upgrades the HTTP connection into a full two-way connection. Therefore, MQTT combined with WebSocket emerges as the recommended efficient protocol for the Internet of Things.

The Aggregation / Bus Layer aggregates and combines communications from different devices and routes communications as a gateway to specific devices. Additionally, the aggregation / bus layer is responsible for bridging and transformations between protocols and supports a HTTP server and a MQTT broker.

The Event Processing and Analytics Layer [21] are responsible for analyzing events, which are taken from the bus and stored into a database. There are different approaches to be used in the Event Processing and Analytics Layer: scalable column-based data storage, map-reduce for long-running batch-oriented data processing, complex event processing for fast in-memory processing, and traditional application server processing.

The External Communication Layer [13], [21] enables communication outside of devices by supporting processing models like: Web-based frontends and portals, dashboards with analytics processing views, and system interaction outside the network via APIs. The Device Management Layer contains the Device Manager component and related device manager agents for different platform and device types. The device manager is responsible for the installed software, enabling and disabling features of devices, managing security controls and identifiers, monitoring the availability of devices, and locking the device remotely.

A current holistic approach for the development of the Internet of Things environments is presented in [13]. This research has a close link to our work about leveraging the integration of the Internet of Things into a framework of digital enterprise architectures. The main contribution from [13] considers a role-specific development methodology and a development framework for the Internet of Things. The development framework contains a set of modeling languages for a vocabulary language to describe domain-specific features of an IoT-application, an architecture language for describing application-specific functionality, and a deployment language for deployment features.

## 4 Digital Enterprise Architecture

Enterprise Architecture Management (EAM) [7] defines today with frameworks, standards [11], [12], tools and practical expertise a quite large set of different views and perspectives. We argue in this paper that a new and refocused digital enterprise architecture approach should support digitization of products and services and should be both holistic [37] and [6] and easily adaptable [18], [3] to support the digital transformation. We are evolving the first versions of ESARC–Enterprise Services Architecture Reference Cube [22], [23] (Fig. 3).

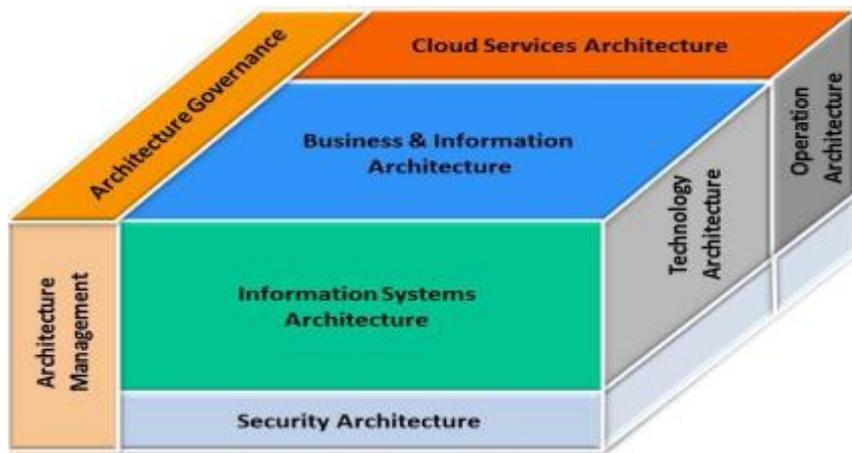
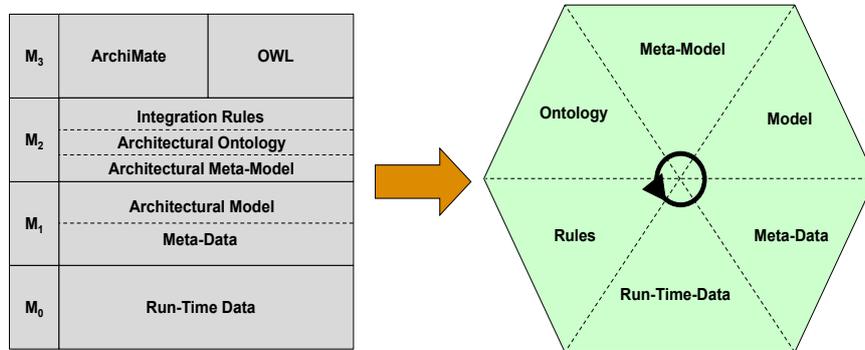


Fig. 3. Enterprise Services Architecture Reference Cube [22], [23]

In this paper we extend our service-oriented enterprise architecture reference model for the context of managed architectural cases with decision making [6], [24] which are supported by interactive functions of an EA cockpit [5]. Additionally, we have tailored our architectural metamodel integration approach [23] to support architectures for digital transformations with Microservices and Internet of Things.

ESARC is more specific than existing architectural standards of EAM – Enterprise Architecture Management [18] and [19] and extends these architectural standards for digital enterprise architectures with services and cloud computing. ESARC provides a holistic classification model with eight integral architectural domains. These architectural domains cover specific architectural viewpoint descriptions in accordance to orthogonal dimensions of both architectural layers and architectural aspects [19]. ESARC abstracts from a concrete business scenario or technologies, but it is applicable for concrete architectural instantiations to support digital transformations. The Open Group Architecture Framework [18] provides the basic blueprint and structure for our extended service-oriented enterprise architecture domains of ESARC [22] having: Architecture Governance, Architecture Management, Business and Information Architecture, Information Systems Architecture, Technology Architecture, Operation Architecture, and Cloud Services Architecture.

Our research extends a previous metamodel-based model extraction and integration approach from [23] for digital enterprise architecture viewpoints, models, standards, frameworks and tools to support the adaptable integration of micro-granular architecture. Currently we are working on the idea of continuously integrating small architectural descriptions (Fig. 4) for relevant objects of a digital enterprise architecture. To continuously integrate a huge amount of dynamically growing architectural descriptions from different microstructures with micro-granular architecture into a consistent enterprise architecture is a considerable challenge. In order to address this problem, we are currently formalizing small-decentralized mini-metamodels, models, and data of architectural microstructures, like Microservices and IoT into EA-Mini-Descriptions. From the case of a web shop we can extract the following micro-granular structure examples with their local architecture models: OrderService and BillingService.



**Fig. 4.** Structure of EA-Mini-Descriptions

EA-Mini-Descriptions consists of partial EA-Data, partial EA-Models, and partial EA-Metamodels associated with Microservices and/or Internet of Things. These structures are based on the Meta Object Facility (MOF) standard [10] of the Object Management Group (OMG). The highest layer M3 represents the abstract language concepts used in the lower M2 layer and is therefore the meta-metamodel layer. The next layer M2 is the metamodel layer and defines the language entities for M1 (e.g. constructs from UML, ArchiMate [12], or OWL [23]). Instantiations of these languages then form the layer M1 that contains models in the specified language. These models are a structured representation of the lowest layer M0 that is formed by collected concrete data from real-world use cases.

For building our EA-Mini-Descriptions, we applied the four layers of MOF to provide sufficient information structures for an EA integration scenario with microstructures. M0 and M1 are local layers to a single microstructure (*cell* metaphor). While M0 consists of operational run-time or monitoring data, M1 contains meta-data of the microstructure (e.g. purpose, API endpoints, or usage costs) as well as its inner architectural model (e.g. components or communication channels). On top of these, the layer M3 acts as a global meta-model layer that holds necessary information for several collaborating microstructures (*body* metaphor, combining several *cells*). It incorporates architectural meta-models and ontologies of micro-granular systems and services while also providing the important integration rules for

the semi-automatic integration of specific metamodels to the overall integrated and dynamically growing EA metamodel from the composition of EA-Mini-Descriptions. On top of that, M3 specifies the languages and semantic representations that we are using for modeling and representing adaptable enterprise architecture metamodels.

Adaptability in the context of EA and microstructures is mostly concerned with heterogeneity, distribution, and volatility. Adaptation [3] is a key success factor for the survival of digital enterprise architectures, platforms, and application environments. Therefore, we have extracted the idea of digital *ecosystems* from [18] and linked this with main strategic drivers for system development and system evolution. Additionally, we have to consider internal factors: The alignment of Architecture Governance [3] shapes resiliency, scalability, and reusability of components and services for distributed information systems.

## 5 Conclusion

In this paper we identified the need for a bottom-up integration of a huge amount of dynamically growing micro-granular systems and services, like Microservices and the Internet of Things, as part of a new suited digital enterprise architecture. In order to integrate micro-granular architecture models for a living and holistic digital enterprise architecture model we are extending more traditional enterprise architecture reference models with state of art elements for agile architectural engineering to support the digitization of products, services, and processes.

According to our research questions we have leveraged a new enterprise architecture approach to model a living digital enterprise architecture, which is well in line with adaptive models and digital transformation mechanisms. We have investigated new architectural properties of Microservices as a base for integrating them into our digital enterprise reference architecture. We have extended in our work the new architectural integration context from the Internet of Things architecture to support Microservices as well and the digital transformation of products and services. Finally, we have extended our previous quite static enterprise architecture reference model to be able to integrate micro-granular systems and services, like Microservices and Internet of Things. This is a fundamental extension of our previous work on the ESARC reference model to be able to integrate through a continuously bottom-up approach a huge amount of micro-granular systems with own and heterogeneous local architectures.

We have to additionally consider alternative approaches for the integration of a large set of divergent systems considering an open world approach. Our approach has some limitations of our original focus with manually working integration models for existing architectural metamodels assuming a closed world of a classical enterprise.

We are currently working on extended decision support mechanisms for an architectural cockpit for digital enterprise architectures and related engineering processes. Future work will extend both mechanisms for adaptation and flexible integration of digital enterprise architectures as well as decisional processes with rationales and explanations.

## References

1. Bass, L., Weber, I., Zhu, L.: DevOps: A Software Architect's Perspective. Addison Wesley (2015)
2. Evans, E.: Domain-driven Design: Tackling Complexity in the Heart of Software. Addison Wesley (2004)
3. Farwick, M., Pasquazzo, W., Breu, R., Schweda, C. M., Voges, K., Hanschke, I.: A Meta-Model for Automated Enterprise Architecture Model Maintenance. EDOC, 1-10 (2012)
4. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): A vision, architectural elements, and future directions. *F. Gen. Comp. Syst.* 29(7), 1645-1660 (2013)
5. Jugel, D., Schweda, C. M.: Interactive functions of a Cockpit for Enterprise Architecture Planning. EDOCW, Ulm, Germany, 2014, IEEE, 33-40 (2014)
6. Jugel, D., Schweda, C.M., Zimmermann, A.: Modeling Decisions for Collaborative Enterprise Architecture Engineering. In: CAISE / TEAR 2015 Springer, 351-362 (2015)
7. Lankhorst, M.: Enterprise Architecture at Work. Springer (2013)
8. Newman, S.: Building Microservices: Designing Fine-Grained Systems. O'Reilly (2015)
9. Ollife, G.: Microservices: Building Services with the Guts on Outside. (2015), online at <http://blogs.gartner.com/garryollife/2015/01/30/microservices-guts-on-the-outside/>.
10. OMG, Meta Object Facility (MOF) Core Specification, Version 2.5 (2011)
11. Open Group: TOGAF Version 9.1. Van Haren Publishing (2011)
12. Open Group: ArchiMate 2.0 Specification. Van Haren Publishing (2012)
13. Patel, P., Cassou, D.: Enabling High-level Application Development for the Internet of Things. In CoRR abs/1501.05080, *Journal of Systems and Software* (2015)
14. Porter, M. E., Heppelmann, J. E.: How Smart Connected Products are Transforming Competition. *Harvard Business Review*, November, 1-23, (2014)
15. Schmidt, R., Möhring, M., Härting, R. C., Reichstein, C., Neumaier, P., & Jozinović, P.: Industry 4.0-Potentials for Creating Smart Products: Empirical Research Results. In *Business Information Systems*, Springer International Publishing, 16-27, (2015)
16. Schmidt, R., Zimmermann, A., Möhring, M., Nurcan, S., Keller, B., & Bär, F.: Digitization-Perspectives for Conceptualization. In *Advances in Service-Oriented and Cloud Computing* (pp. 263-275). Springer International Publishing, (2015)
17. Tiwana, A.: Platform Ecosystems. Morgan Kaufmann (2013)
18. Trojer, T. et. al.: Living Modeling of IT Architectures: Challenges and Solutions. *Software, Services, and Systems 2015*, 458-474 (2015)
19. Westerman, G., Bonnet, D.: Revamping Your Business Through Digital Transformation. *MIT Sloan Management Review*, February (2015)
20. Weill, P., Ross, J. W.: IT Governance: How Top Performers Manage It Decision Rights for Superior Results. Harvard Business School Press (2004)
21. WSO2 White Paper: Reference Architecture for the Internet of Things. Version 0.8.0 <http://wso2.com> (2015)
22. Zimmermann, A., Gonen, B., Schmidt, R., El-Sheikh, E., Bagui, S., Wilde, N.: Adaptable Enterprise Architectures for Software Evolution of SmartLife Ecosystems. EDOCW, IEEE, 316-323 (2014)
23. Zimmermann, A., Schmidt, R., Sandkuhl, K., Wißotzki, M., Jugel, D., Möhring, M.: Digital Enterprise Architecture – Transformation for the Internet of Things. EDOC 2015 SoEA4EE, 21-25 September 2015, Adelaide, Australia, IEEE, 130-138 (2015)
24. Zimmermann, A., Jugel, D., Sandkuhl, K., Schmidt, R., Schweda, C., Möhring, M.: Architectural Decision Management for Digital Transformation of Products and Services. *Complex Systems Informatics and Modeling Quarterly, CSIMQ* 2016, No. 6, 31-53 (2016)