RC25624 (WAT1609-040) September 13, 2016
Computer Science

IBM Research Report

Proceedings of the 10th Advanced Summer School on Service Oriented Computing

Johanna Barzen[1], Rania Khalaf[2],
Frank Leymann[1], Bernhard Mitschang[1], Editors

[1]University of Stuttgart
Germany

[2]IBM Research Division
One Rogers Street
Cambridge, MA 02142-1203
USA

Research Division
Almaden – Austin – Beijing – Brazil – Cambridge – Dublin – Haifa – India – Kenya – Melbourne – T.J. Watson – Tokyo – Zurich

The 10th Advanced Summer School on Service-Oriented Computing

June 27 – July 1
2016
Hersonissos, Crete, Greece

The 10th Advanced Summer School on Service Oriented Computing (SummerSOC'16) continued a successful series of summer schools that started in 2007, regularly attracting world-class experts in Service Oriented Computing to present state-of-the-art research during a week-long program organized in several thematic tracks: patterns, formal methods for SOC, computing in the clouds, data science, e-Health and emerging topics. The advanced summer school is regularly attended by top researchers from academia and industry as well as by graduate students from programs with international acclaim, such as the Erasmus Mundus International Master in Service Engineering.

During the morning sessions at SummerSOC renowned researchers gave invited tutorials on subjects from the themes mentioned above. The afternoon sessions were dedicated to original research contributions in these areas: these contributions have been submitted in advance as papers that had been peer-reviewed. Accepted papers were presented during SummerSOC and during the poster session. Furthermore, PhD students had been invited based in prior submitted and reviewed extended abstracts to present the progress on their theses and to discuss it during poster sessions. Some of these posters have been invited to be extended as a full paper, which are included in this Technical Report.

Also, this year the "Christos Nikolaou Memorial Ph.D. Award" to honor Prof. Christos Nikolaou's career-long contributions in university education and research was established. The first winner of this Christos Nikolaou Memorial Ph.D. Award is Jörg Lenhard, who presents the core ideas of his awarded thesis in this Technical Report too. The award is not only an honor and distinction but is associated with 2000€ for the awardee, sponsored by StartTech Ventures.

Johanna Barzen, Rania Khalaf, Frank Leymann, Bernhard Mitschang
- Editors -

Adaptable Digital Enterprise Architecture with Microservices

Justus Bogner[1, 2] and Alfred Zimmermann[1]

Herman Hollerith Center, Boeblingen, Germany[1]
Hewlett Packard Enterprise, Boeblingen, Germany[2]
justus.bogner@hpe.com, alfred.zimmermann@reutlingen-university.de

1 Introduction and Motivation

The fast moving process of digitization[1] demands flexibility in order to adapt to rapidly changing business requirements and newly emerging business opportunities. New features have to be developed and deployed to the production environment a lot faster. To be able to cope with this increased velocity and pressure, a lot of software developing companies have switched to a Microservice Architecture (MSA) approach. Applications built this way consist of several fine-grained and heterogeneous services that are independently scalable and deployable. However, the technological and business architectural impacts of microservices based applications directly affect their integration into the digital enterprise architecture. As a consequence, traditional Enterprise Architecture Management (EAM) approaches are not able to handle the extreme distribution, diversity, and volatility of micro-granular systems and services. We are therefore researching mechanisms for dynamically integrating large amounts of microservices into an adaptable digital enterprise architecture.

2 Microservices and Adaptable Enterprise Architecture

The term microservices became popular around 2013 and refers to a fine-grained style of service-oriented architecture (SOA). James Lewis defines a Microservice Architecture as an approach for building a single application as a set of small independent services. [2] Each of these services runs in its own process and communicates with lightweight mechanisms. Microservices are built around business capabilities, are independently deployable, and may utilize very different 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. Using MSAs, organizations can expand agility and flexibility for business and IT systems. However, microservices also come with the

[1] Westerman, G. et al: Leading Digital: Turning Technology Into Business Transformation. Harvard Business Press, 2014
[2] Newman, S.: Building Microservices: Designing Fine-Grained Systems, O'Reilly, 2015

need for a strong DevOps culture to handle the increased distribution level and deployment frequency. Moreover, while each single microservice is of low complexity, the overall complexity of the system has not been reduced. It has moved from the inner architecture to the outer architecture (see Fig. 1).

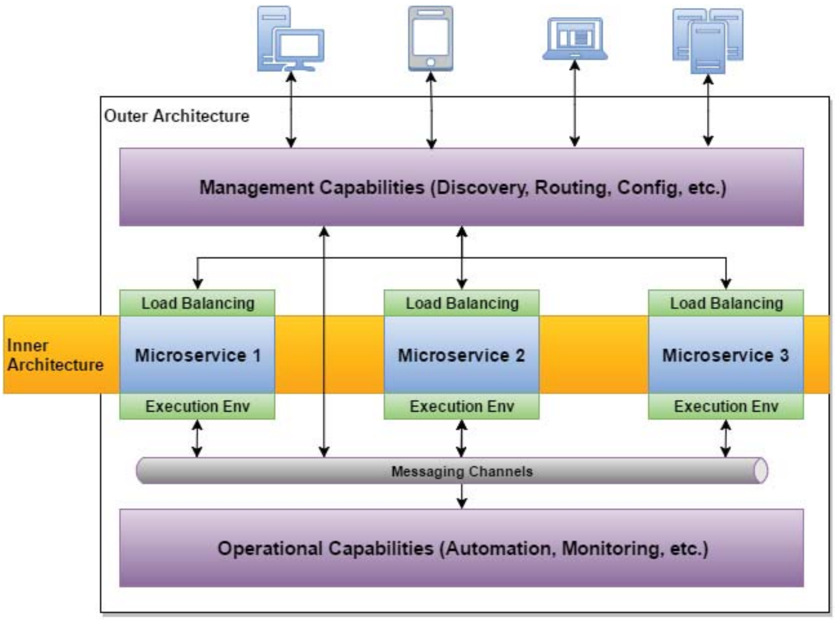

Fig. 1. Inner and Outer Architecture of Microservices[3]

In the context of EAM, challenges with microservices are mostly concerned with heterogeneity, distribution, and volatility. To integrate a huge amount of dynamically growing architectural descriptions of very different microservices into a consistent enterprise architecture is a considerable challenge. In order to tackle this problem, we are currently working on the formalization of small-decentralized mini-metamodels, models, and data of architectural microservice descriptions based on the Meta Object Facility standard[4] (EA-Mini-Descriptions, see Fig. 2). M0 as well as M1 serve as local layers to a single microservice (*cell* metaphor) with M0 representing operational run-time or monitoring data and M1 providing metadata (e.g. purpose, API endpoints, or usage costs) and also its inner architectural model (e.g. components or communication channels). Using these two as a foundation, M2 works as a global metamodel layer with information for several communicating microservices (*body* metaphor, combining several *cells*). It holds architectural metamodels and ontologies while providing important integration rules for semi-automatic integration. These metamodels are then included

[3] Based on Olliffe, G.: Microservices: Building Services with the Guts on the Outside, 2015. Retrieved March 18, 2016 from http://blogs.gartner.com/gary-olliffe/2015/01/30/microservices-guts-on-the-outside

[4] OMG, 2011, OMG Meta Object Facility (MOF) Core Specification, Version 2.5

into the holistic and dynamically growing EA metamodel from the composition of EA-Mini-Descriptions. Finally, layer M3 defines the semantic representations and languages that are used for modeling and representing these adaptable enterprise architecture metamodels.

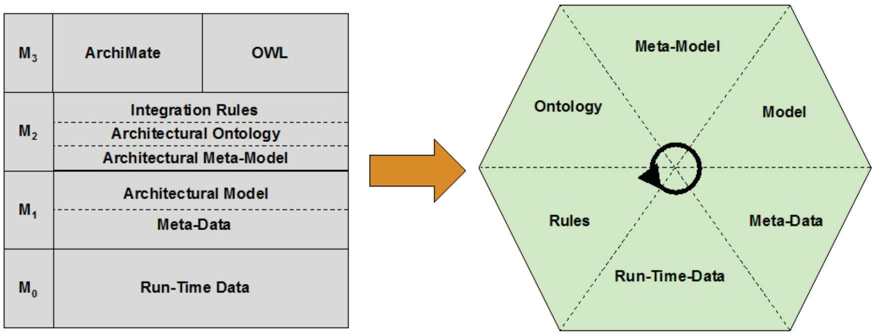

Fig. 2. Structure of EA-Mini-Descriptions

As a next step, the presented EA-Mini-Descriptions are used with the Enterprise Services Architecture Model Integration (ESAMI) method[5] to perform correlation analysis, which provides an instrument for systematic integration. The iterative approach is based on special correlation matrices handled by a manual process to identify similarities between analyzed model elements. The chosen elements are then integrated according to their most valuable contribution towards a holistic reference model. This continuous model refinement allows to integrate even extremely heterogeneous microservices that may in fact not share a complete metamodel.

3 Conclusion and Future Work

The presented architectural properties of microservices demand advanced Enterprise Architecture methodologies for the integration of structures with a micro-granular architecture into an overall adaptable EA. Our EA-Mini-Descriptions can serve as flexible metamodels for microservices that can be combined into larger entities. Through a manual correlation-based model analysis and integration approach, we presented means to merge these microservices into a holistic, but dynamically adjusting reference architecture. Future research may include the automatic machine-supported creation of our EA-Mini-Descriptions (at least partially). Similarly, it may be of interest to support the manual integration decision by automated systems, e.g. via mathematical comparisons (similarity, Euclidean distance), semantic integration rules, or data analytics and data mining techniques. These methods can significantly ease associated manual efforts and reduce the rate of architectural errors in traditional EA models and approaches.

---

5 Zimmermann, A. et al: Towards an integrated service-oriented reference enterprise architecture, Proc. 2013 Int. Work. Ecosyst. Archit. - WEA 2013, pp. 26–30, 2013