Automatic Migration to Microservice: A Model-Driven Approach

Antonio Bucchiarone[1], Kemal Soysal[2], Claudio Guidi[3]

[1] Fondazione Bruno Kessler, Trento, Italy
bucchiarone@fbk.eu
[2] LS IT-Solutions GmbH, Berlin, Germany
kemal.Soysal@ls-it-solutions.de
[3] italianaSoftware s.r.l., Imola, Italy
cguidi@italianasoftware.com

Extended Abstract

Microservices [1] is an architectural style originating from Service-Oriented Architectures (SOAs) [2]. The main idea is to structure systems by composing small independent building blocks communicating exclusively via message passing. These components are called *microservices*. The characteristic differentiating the new style from monolithic architectures and classic Service-Oriented is the emphasis on *scalability*, *independence*, and *semantic cohesiveness* of each unit constituting the system. Migration from monolithic applications (MA) to microservices-based application (MSA) is a challenging task that very often it is done manually by the developers taking into account the main business functionalities of the input application and without a supporting tool.

In this talk, we present a model-driven approach for the automatic migration to microservices. The approach is implemented by means of JetBrains MPS[1], a text-based metamodelling framework, and validated using a first migration example from a Java-based application to Jolie[2] - a programming language for defining microservices.

A first version of the proposed approach has been already published in [3] and shows how a MA can be converted into a MSA and deployed in a Docker container. The main contribution is visualized in Figure 1, which depicts the different artifacts realized and their relations. Technically, the solution is composed by two fundamental components: (a) the `Microservices Miner` and (b) the `Microservices Generator`, by two Domain Specific Languages (DSLs) (i.e., for the `Microservice` specification and for their `Deployment`), by a set of `generators` used to support the overall migration from MA (developed in Java) to MSA (developed using Jolie) and the corresponding deployment in a `Docker Container`.

A prototype implementation of the approach and the related artefacts are available at the GitHub repository: `https://github.com/antbucc/Migration.git`. Using our solution, a developer can start the migration process importing the Java source code of the monolithic application in phase (1). This is done using the native MPS action `Get Models Content from Source` that can be invoked by the main menu. In this way the Java code is parsed into MPS' base language and imported in the editor as input Java models. Phase (2) is used to interrogate the imported Java models for patterns of used packages, classes, methods and members to identify microservice candidates. In phase (3), with the `Microservice` domain specific language in the hand the developer can create the different models of the identified microservices. In phase (4), using the provided generators, the microservices models with their respective interfaces

[1] `https://www.jetbrains.com/mps/`
[2] `https://www.jolie-lang.org/`

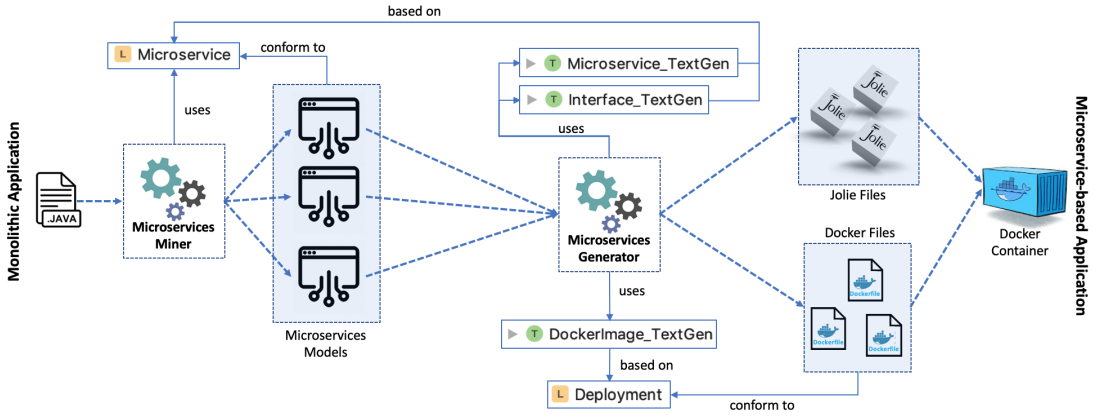

Figure 1: The Model-Driven Migration Approach.

are transformed in the target Jolie files. In the end, in phase (5), for each Jolie microservice a `Dockerfile` is created and used to deploy the overall application in a Docker container.

The framework have been evaluated with an simple scenario with the aim to demonstrate the feasibility of the approach, and calls for future research. To make it scalable and usable in real contexts we are interested to test it using different industrial case studies to further investigate the soundness of the proposed methodology and eventually to extend the specification of the provided DSLs in MPS to make it more general.

References

[1] N. Dragoni, S. Giallorenzo, A. Lluch-Lafuente, M. Mazzara, F. Montesi, R. Mustafin, and L. Safina. Microservices: yesterday, today, and tomorrow. In *Present and Ulterior Software Engineering*. Springer, 2017.

[2] M.C. MacKenzie et al. Reference model for service oriented architecture 1.0. *OASIS Standard*, 12, 2006.

[3] Antonio Bucchiarone, Kemal Soysal, and Claudio Guidi. A model-driven approach towards automatic migration to microservices. In Jean-Michel Bruel, Manuel Mazzara, and Bertrand Meyer, editors, *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment - Second International Workshop, DEVOPS 2019, Château de Villebrumier, France, May 6-8, 2019, Revised Selected Papers*, volume 12055 of *Lecture Notes in Computer Science*, pages 15–36. Springer, 2019.