Model-Based Migration of Legacy Software Systems into the Cloud: The CloudMIG Approach

Sören Frey and Wilhelm Hasselbring

Software Engineering Group
University of Kiel, 24118 Kiel
{sfr, wha}@informatik.uni-kiel.de

Abstract

Cloud computing supplies software, platforms, and infrastructures as a service (abbr. SaaS, PaaS, and IaaS, respectively) over a network connection. From a user's perspective, resource efficiency and scalability can be improved through aligning a cloud-based application with the cloud environment. Running an existing software system on a cloud computing basis usually involves extensive reengineering activities during the migration. We present identified shortcomings of current approaches and introduce our model-based approach CloudMIG for the semi-automated migration of enterprise software systems to scalable and resource-efficient PaaS and IaaS-based applications which addresses these shortcomings.

1 Introduction

Newly developed enterprise software may easily be designed for utilizing cloud computing technologies [1] in a greenfield project. In contrast to that, running an existing software system on a cloud computing basis usually involves extensive reengineering activities during the migration. The actual software architecture of that system often has to be reconstructed and an architecture acting as a target for a transformation step has to be built, for instance. Nevertheless, instead of recreating the functionalities of a system from scratch for being compatible with a selected cloud environment, a migration enables the reuse of substantial parts of a system. The rest of the paper is structured as follows. Section 2 states the identified shortcomings of current approaches. Section 3 presents the approach CloudMIG which addresses these shortcomings, before Section 4 draws the conclusions.

2 Shortcomings of Current Approaches

Today's projects migrating typical enterprise software to cloud-based applications suffer several shortcomings. They can be summarized as follows:

S1 - Applicability: Solutions for migrating enterprise software to cloud-based applications are limited to particular cloud providers.

S2 - Level of automation: The target architecture and the mapping model often have to be built entirely manual. Additionally, the target architecture's violations against the cloud environment's imposed constraints (e.g. restrictions concerning the creation of network sockets or limitations of programming interfaces) are not identified automatically at design time.

S3 - Resource efficiency: Various migrated software systems are not designed to be resource-efficient and do not leverage the cloud environments' elasticity, because even transfering an established application to a new cloud environment is a challenging task itself. Furthermore, means for evaluating a target architecture's dynamic resource utilization at design time are most often inadequate.

S4 - Scalability: Automated support for evaluating a target architecture's scalability at design time is rare in a cloud computing context.

3 The Approach CloudMIG

CloudMIG is composed of six activities for migrating an enterprise system to PaaS and IaaS-based cloud environments [2]. It provides model-driven generation of considerable parts of the system's target architecture and fosters resource efficiency and scalability on an architectural level. Figure 1 outlines the approach. Its activities A1-A6 are briefly described in the following including the involved models.

A1 - Extraction: A model describing the actual architecture of the legacy system is extracted by means of a software architecture reconstruction methodology. We propose OMG's KDM[1] for building a suitable metamodel. For leveraging the commonly applied utility computing paradigm, the target architecture has to be laid out resource-efficient and elastic. Therefore, CloudMIG includes the extraction of an established software system's utilization model acting as a starting point. The utilization model includes statistical properties concerning user behavior like service invocation

[1] http://www.omg.org/spec/KDM/

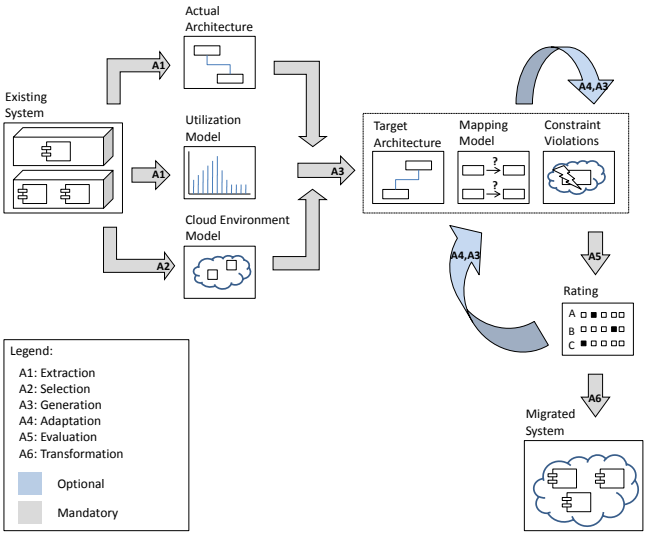

Figure 1: CloudMIG Overview

rates over time. Furthermore, the utilization model contains application-inherent information related to proportional resource consumption. We propose OMG's SMM[2] as a foundation for building the related metrics meta-model.

A2 - Selection: Common properties of different cloud environments are described in a cloud environment meta-model. Selecting a provider-specific cloud environment as a target platform for the migration activities therefore implies the existence and selection of a specific instance of this meta-model.

A3 - Generation: The generation activity produces three artefacts, namely a target architecture, a mapping model, and a model characterizing the target architecture's violations of the cloud environment constraints. The latter lists the features of the target architecture which are non-conform with the cloud environment's specification. The target architecture is realized as an instance of the cloud environment meta-model. We propose three phases P1-P3 for the generation of the target architecture.

P1 *Model transformation:* The phase P1 produces an initial assignment from features of the existing architecture to cloud-specific features available in the cloud environment model.

P2 *Configuration:* The phase P2 serves as a configuration of the algorithm used for obtaining a resource-efficient feature allocation in phase P3. During P2, a reengineer may adjust and prioritize rules and assertions for heuristic computation (cf. P3). A rule could be formulated like the following example: "Distribute the 5 most frequently used services to own virtual machines".

[2] `http://www.omg.org/spec/SMM/1.0/Beta1/`

P3 *Resource-efficient feature allocation:* The phase P3 improves the initial assignment of architectural features generated in phase P1 referring to resource-efficiency. Therefore, the formulated rules are utilized and the compliance of the resulting architecture with the defined assertions is considered. There exists an enormous number of possible combinations for assigning architectural features. Efficiency improvements for one resource can lead to degradation for other resources or impair some design quality attributes. Therefore, we propose application of a heuristic rule-based approach to achieve an overall improvement.

A4 - Adaptation: The activity A4 allows the reengineer to adjust the target architecture manually towards case-specific requirements that could not be fulfilled during generation activity A3. For example, the generation process might not have yielded an expected assignment of a critical component.

A5 - Evaluation: This activity evaluates the outcomes of the activities A3 and A4. The evaluation involves static and dynamic analyses of the target architecture. Considering the target architecture's expected runtime behavior, we plan to incorporate a simulation on the basis of the CloudSim framework [3].

A6 - Transformation: This activity comprises the manual transformation of the enterprise system towards the aimed cloud environment according to the generated and improved target architecture.

4 Conclusion

We presented early work concerning our model-based approach CloudMIG for migrating legacy software systems to scalable and resource-efficient cloud-based applications. It concentrates on the migration of enterprise software systems towards PaaS and IaaS-based cloud environments. To raise the level of automation, CloudMIG is intended to generate considerable parts of a resource-efficient target architecture.

References

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the Clouds: A Berkeley View of Cloud Computing. Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, Feb 2009.

[2] Bhaskar Prasad Rimal, Eunmi Choi, and Ian Lumb. A Taxonomy and Survey of Cloud Computing Systems. *Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC*, pages 44–51, 2009.

[3] Rodrigo N. Calheiros, Rajiv Ranjan, César A. F. De Rose, and Rajkumar Buyya. CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services. *CoRR*, abs/0903.2525, 2009.