

# Chapter 4

## From Software Specification to Cloud Model

**Dušan Savić**

*University of Belgrade, Serbia*

**Siniša Vlajić**

*University of Belgrade, Serbia*

**Marijana Despotović-Zrakić**

*University of Belgrade, Serbia*

### ABSTRACT

*Applications are often multi-tier and require application servers, workflow engines, and database management systems. Cloud computing is a computing paradigm wherein the resources such as processors, storage, and software applications are provided as services via the Internet. Moving an enterprise application to the cloud can be a challenge. This application needs to be split into the components that then automatically deploy on the cloud. In this chapter, the authors introduce a way to automatically derivate the main architecture components from the software requirements that can serve as a basis for an architecture diagram in the MOCCA method. The proposed approach is model and use case driven.*

### 1. INTRODUCTION

Applications today are often composite, multi-tier applications, consisting of application components such as UIs, services, workflows and databases as well as middleware components such as application servers, workflow engines and database management systems (Mietzner, Unger, & Leymann, 2009). When moving such a composite application into the cloud, decisions

must be made about putting which tier and even which component of such an application to which cloud (Leymann, Fehling, Mietzner, Nowak, & Dustdar, 2011; Andrikopoulos, Binz, Leymann, & Strauch, 2013; Cardoso, Binz, Breitenbücher, Kopp, & Leymann, 2013; Demont, Breitenbücher, Kopp, Leymann, & Wettinger, 2013).

The authors (Leymann, Fehling, Mietzner, Nowak, & Dustdar, 2011) defined Move-to-Cloud problem as a two main problems:

1. How to rearrange the components of a multi-tier, multi-component application into disjoint groups of components, and
2. How each group can be provisioned separately to different clouds.

They describe methodology that allows application developers and architects to model their application components and properties and is named as MOCCA method.

We have developed Silab approach whose the main goal was to enable automated analysis and processing of software requirements in order to achieve automatic generation of different parts of a software system. Silab approach has been used for the Kostmod 4.0 project (Kostmod 4.0, 2009), which was implemented for the needs of the Royal Norwegian Ministry of Defense.

In this chapter we have introduced a way how Silab approach can be integrated with MOCCA approach. Therefore, we have defined transformation which automatically generates an architecture model for MOCCA method from software requirements.

This chapter is organized as follows. In the next section it gives an overview of the cloud computing. Section 3 introduces model and use case development (Jacobson, Christerson, Jonsson, & Overgaard, 1993; Cockburn, 2000), and Section 4 describes Silab approach. Finally, Section 5 presents integrated Silab and MOCCA approach and Section 6 concludes the chapter and outlines some future works.

## **2. CLOUD COMPUTING**

The term cloud was first used at the end of the last century to refer to large ATM networks. At the beginning of this century, with the advent of Amazon's Web Services (Maggiani, 2009) the term cloud computing begins to be widely used (Leymann, 2009; Leimeister, Böhm, Riedl, & Krcmar, 2010; Talukder, Zimmerman, & Prahalad,

2010). Eric Schmidt (Aymerich, Fenu, & Surcis, 2008) was one of the pioneers and promoter of cloud computing. Cloud computing has emerged from grid computing and it is related with technologies and concepts such as: Simple Object Access Protocol (SOAP), Service Oriented Architecture (SOA), Web services, Software-as-a-Services (SaaS) (Janssen & Joha, 2011).

Cloud computing has evolved from Grid computing. The idea of grid computing is a generalization of the Web. Web is primarily a service that allows the presentation of information and exchange data via the Internet. GRID is a service primarily for the sharing of processing power and memory of computers on the network allowing local or global network turns into a huge computer resource.

The main characteristics of Grid Computing (Foster, Kesselman, & Tuecke, 2001) are:

1. Decentralized resource control,
2. Standardization, i.e. open and common protocols and interfaces are used in Grid middleware, and
3. Not-trivial qualities of services.

Grid application requires a middleware to enable communication via open and standardized protocols. All middleware attempts to apply the standards defined by the OGF community (OGF, 2013).

The large computing and storage capacity offered by grid technology has led to the development of another category of services that later came to be identified as cloud services (Aymerich, Fenu, & Surcis, 2008). There are many definition of the cloud computing. For Boss et al. (2007) a Cloud is a pool of virtualized computer resources, complement to Grid environment by supporting the management of Grid resources. Cloud computing implies component-based application construction (Skillicorn, 2002), based on existing technologies but integrated these technologies different from Grid, especially the combination with Utility Computing and data centers (Aaron, 2007).

19 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/from-software-specification-to-cloud-model/102405](http://www.igi-global.com/chapter/from-software-specification-to-cloud-model/102405)

## Related Content

---

### Navigating the Security Maze: Fortifying Cloud Computing Against Evolving Threats

Menard Musendekwa (2025). *Risk-Based Approach to Secure Cloud Migration* (pp. 301-324).

[www.irma-international.org/chapter/navigating-the-security-maze/375542](http://www.irma-international.org/chapter/navigating-the-security-maze/375542)

### A New Conception of Load Balancing in Cloud Computing Using Tasks Classification Levels

Fahim Youssef, Ben Lahmar El Habib, Rahhali Hamza, Labriji El Houssine, Eddaoui Ahmed and Mostafa Hanoune (2018). *International Journal of Cloud Applications and Computing* (pp. 118-133).

[www.irma-international.org/article/a-new-conception-of-load-balancing-in-cloud-computing-using-tasks-classification-levels/213992](http://www.irma-international.org/article/a-new-conception-of-load-balancing-in-cloud-computing-using-tasks-classification-levels/213992)

### Cost-Effective Resource Configurations for Multi-Tenant Database Systems in Public Clouds

Rizwan Mian, Patrick Martin, Farhana Zulkernine and Jose Luis Vazquez-Poletti (2015). *International Journal of Cloud Applications and Computing* (pp. 1-22).

[www.irma-international.org/article/cost-effective-resource-configurations-for-multi-tenant-database-systems-in-public-clouds/127102](http://www.irma-international.org/article/cost-effective-resource-configurations-for-multi-tenant-database-systems-in-public-clouds/127102)

### Secure Mobile Multi Cloud Architecture for Authentication and Data Storage

Karim Zkik, Ghizlane Orhanou and Said El Hajji (2017). *International Journal of Cloud Applications and Computing* (pp. 62-76).

[www.irma-international.org/article/secure-mobile-multi-cloud-architecture-for-authentication-and-data-storage/179538](http://www.irma-international.org/article/secure-mobile-multi-cloud-architecture-for-authentication-and-data-storage/179538)

### Integrating Fog Computing With AI for Real-Time Disaster Management in Smart Cities

Sarvesh Chand (2025). *International Journal of Fog Computing* (pp. 1-19).

[www.irma-international.org/article/integrating-fog-computing-with-ai-for-real-time-disaster-management-in-smart-cities/376243](http://www.irma-international.org/article/integrating-fog-computing-with-ai-for-real-time-disaster-management-in-smart-cities/376243)