

## Dissertation

# Kontraktbasierte Modellierung, Implementierung und Suche von Komponenten in serviceorientierten Architekturen

Marc Lohmann

## Abstract

Service oriented architectures promise an easy integration of existing IT systems inside and outside of an organization. Service descriptions and software developers are the pivotal elements when systems are integrated over a service oriented architecture.

The integration of a service into a client is based on its public service description. On the service provider side, a software developer develops a service description and implements a service accordingly. Today, a service is implemented mainly manual. Automated code generation techniques are seldom used. On the service requestor side, if a software developer needs a specific functionality he searches for a sufficient service. Thus, the success of service oriented architectures is strongly coupled to the content and usability of service descriptions by software developers.

In this thesis we describe how to use contracts on the model level to enable a correct implementation and retrieval of services. In our approach we take the pivotal roles of the software developer and the public service description into account. Especially, we allow for a model-based semantic description, monitoring of the correctness of a service and the comparison of existing service descriptions. Therefore, our approach is based on three important techniques:

*Visual contracts* allow for specifying the semantics of a service. A visual contract consists of a pair of UML composite structure diagrams for the representation of the pre- and post-condition of a service. By using UML composite structure diagrams, we have chosen a graphical notation that is familiar to software developers and easily integrates with today's model-driven software development processes.

By embedding our visual contracts into a software development process we are able to monitor the correctness of a hand-coded implementation. Therefore, we will show how models can be used to generate assertions which monitor the execution of the hand-coded implementation. Herewith, violations of the modeled requirements will be detected at runtime. We call this novel approach *model-driven monitoring*.

*Model-driven Matching* is a new approach to enable a model-based query of existing services. Therefore, we allow for using visual contracts on the service requestor side to describe a needed service. A matching procedure checks the compatibility of a service description and a service query.