Abstract:
UML is a widely adopted open standard to create architectural models from multiple viewpoints for various domains. Its language-inherent extension mechanism is being applied to systematically integrate domain-specific concepts via libraries and profiles because they are indispensable for model-based engineering (MBE). Cloud computing is an appealing target domain for MBE. Modern cloud environments support a relatively high degree of automation in service provisioning, which allows cloud users to dynamically acquire services required for deploying cloud applications. On the other hand, MBE aims at increasing the automation of application development by a systematic tool-supported refinement of high-level models towards a target platform and the environment underneath. The selected platform and environment designate the technical target domain whose concepts have to be captured on the model level in order to enable the refinement of architectural models. Modeling concepts and tools along with a set of constraints on how they can be used denote an architecture style. Providing an architecture style for cloud application modeling based on UML including tools that exploit automated processes of both cloud computing and MBE is thus highly desirable. Due to the generic nature of UML, it does however not provide cloud modeling concepts by default and existing UML tools do not yet adequately support cloud-specific model refinement.
To address these deficiencies, the goal of this thesis is to realize cloud-specific extensions to UML and a toolset that together form an architecture style for developing cloud applications. In particular, we place emphasis on the automation of development processes and their effectiveness in producing truly useful models. Four main contributions are presented to achieve this goal. First we systematically review current cloud modeling languages (CMLs) and investigate major cloud environments to derive a core set of features inherent to the cloud computing domain. They serve as the basis for developing the UML-based cloud application modeling language (CAML), which is the second contribution. CAML supports semi-automatic model refinement towards the Java platform and three major cloud environments via dedicated libraries and profiles. The third contribution addresses the automatic translation of UML architecture models refined by CAML into TOSCA, a recently adopted standard that aims at automating application provisioning and management. Combining UML and TOSCA closes the gap between architecture modeling and application provisioning. As model transformations are key to automate the refinement and translation of model-based artifacts, maintaining those transformations and their produced artifacts is addressed by the fourth contribution. We exploit incremental transformation to co-evolve existing models with changes to transformations. In addition to the conceptual contributions, we provide proof-of-concept implementations as open-source projects and present case studies for evaluating not only their practical relevance but also aspects such as quality and performance.