ABSTRACT

Over the last years, Web services have become increasingly popular. It is because they allow businesses to share data and business process logic through a programmatic interface across networks. In order to reach the full potential of Web services, they can be combined to achieve specific functionalities.

Web services run in complex contexts where arising events may compromise the quality of the system (e.g. a sudden security attack). As a result, it is desirable to count on mechanisms to adapt Web service compositions (or simply called *service compositions*) according to problematic events in the context. Since critical systems may require prompt responses, manual adaptations are unfeasible in large and intricate service compositions. Thus, it is suitable to have autonomic mechanisms to guide their self-adaptation. One way to achieve this is by implementing variability constructs at the language level. However, this approach may become tedious, difficult to manage, and error-prone as the number of configurations for the service composition grows.

The goal of this thesis is to provide a tool-supported framework to guide autonomic adjustments of context-aware service compositions using models at runtime. This framework spans over design time and runtime to face arising known and unknown context events (i.e., foreseen and unforeseen at design time) in the closed and open worlds, respectively.

At design time, we propose to create the models that guide autonomic changes. In order to reach optimum adaptations, a variability model and its possible configurations are verified at design time. At runtime, when problematic events arise in the context, the variability model is leveraged for guiding autonomic changes of the service composition. The activation and deactivation of features in the variability model result in changes in a composition model that abstracts the underlying service composition. Changes in the variability model are reflected into the service composition by adding or removing fragments of Web Services Business Process Execution Language code, which are deployed at runtime.

Under the closed-world assumption, the possible context events are fully known at design time. These events will eventually trigger the *dynamic adaptation* of the service composition. Nevertheless, it is difficult to foresee all the possible situations arising in uncertain contexts where service compositions run. Therefore, the proposed framework also covers the *dynamic evolution* of service compositions to deal with unexpected events in the open world. If dynamic adaptations are not enough to solve uncertainty, the supporting models self-evolve according to abstract tactics, which preserve expected requirements.

The proposal has been validated with a case study and simulations. The answers to several research questions demonstrate the feasibility of models at runtime to guide dynamic adjustments of autonomic service compositions in the closed and open worlds.