Architectural Style-based Modeling and Simulation of Middleware for Mobile Systems

Ping Guo

Today, mobility is one of the most important market and technology trend within information and communication technology. As the demand for rapid deployment of dependable mobile applications increases, middleware for mobile systems is emerging as one of the most active areas of system research in mobility. The middleware is a set of distributed software services that exist between distributed operating systems and mobile applications. The key to the middleware is to provide support across the mobile application domains, help application developers overcome the complexity and problems brought by mobility, and enhance dependability and usability of developed mobile applications. The criticality and pervasiveness of middleware for mobile systems is continually growing. However, the design and development of the middleware are difficult tasks, and it is not easy to ensure the quality of a developed middleware. This is mainly caused by the increasing complexity of the middleware. In addition, the great diversity of this area makes it very difficult for the designers to reuse the already established design knowledge or successful experience when building new systems. All these make the design process quite inefficient and unpredictable, and therefore risking the project.

"One man's magic is another man's engineering". Engineering design is much more routine than innovative. Founding on this fundamental notion in software engineering, we develop an architectural style based approach to deal with the problems in the thesis. We build architectural styles for a class of related middleware. The style represents a common form of design, which originates from the results that practitioners have achieved in one area. The style is formulated to repeat successes and avoid failures from previous projects. When building a new middleware, the designers and developers do not need to explore all possible alternatives for its supported architecture. Instead, they can use the architectural style that is effective for the middleware. They can define the design as instances of the style, or they can use the style as a reference model for further improvement and development. By structuring the design space for a family of related middleware, the style can drastically simplify the process of building a middleware, reduce costs of implementation through reusable infrastructure, and improve system integrity through style-specific analysis and checks.

We develop the approach based on UML-like meta modeling and graph transformation techniques to support sound methodological principals, powerful modeling, formal analysis and refinement. The approach consists of several main parts: the modeling language that supports specification of the style and mobility, the refinement formalization that ensures that an abstract style is correctly refined to a concrete one, the consistency check framework that validates behavioral consistency between two styles on different abstract layers. With the Fujaba simulation tool support, we also develop a style - based engineering process that helps us to efficiently develop correct and consistent styles. Besides, it allows a seamless integration of our approach into the well-known object oriented design. By providing a concrete example of how to construct the style for a class of related middleware, and how to use the style to help the design and development of a new middleware, we show that the architectural style based approach is useful and practical.