Part I: Language and Verification for Collective Autonomic Systems

The first chapters of this book explore foundations for reliable and trustworthy ensembles: languages and verification techniques for individual components, for systems consisting of many individual components, and for the networks and connectors with which components communicate.

The first chapter introduces the SCEL language, a formal language for modeling and programming systems consisting of interacting autonomic components. Each SCEL component contains processes operating on a knowledge repository and is equipped with an interface consisting of attributes that describe the features of the component. Components can dynamically form ensembles based on predicates over interface attributes. Behaviors and interactions in SCEL can be controlled by policies. FACPL is a language for expressing hierarchically-structured, high-level policies. jRESP is a framework that allows Java programs to use the linguistic constructs of SCEL.

The second chapter focuses on foundational aspects of the infrastructure for adaptive systems: networks and reconfigurable connectors. The authors define the Network-Conscious Pi-calculus (NCPi), an extension of the pi-calculus in which network nodes and links are explicitly represented. NCPi can serve as framework for modeling and and verifying systems with programmable network infrastructure, such as peer-to-peer networks. The NCPi calculus is applied to various modeling and verification tasks, e.g., for the PASTRY protocol. The second part of the chapter introduces BIP, the main language for verifying components and ensembles of ASCENS. It also establishes a correspondance between BIP and Petri nets and presents two extensions, reconfigurable and dynamic BIP.

Verification of system properties is an important goal of the ASCENS project, and the third chapter presents various techniques and tools that were developed as part of ASCENS. The techniques comprise qualitative methods that verify Boolean properties, as well as quantitative methods that evaluate a system's performance according to a metric. It is well known that many verification techniques suffer from state explosion: the time or memory to verify a system grows rapidly in the size of its state space. To address this, the chapter stresses the use of compositional verification techniques, in which properties of a system are established based on independent verification of properties of its subsystems. Security aspects are verified using a framework for information-flow analysis that is particularly well suited to checking non-interference and therefore the preservation of information confidentiality.

M. Wirsing et al. (eds.): Collective Autonomic Systems, LNCS 8998, p. 1, 2015.