Towards Architecture-based Autonomic Software Performance Engineering

Xu Zhang*, Chung-Horng Lung*
Greg Franks*

*Carleton University, Ottawa, Canada
{zhangxu, chlung, greg.franks}@sec.carleton.ca

Abstract. Autonomic systems can be self-adaptive and have the potential to achieve high performance through run-time configuration changes. This paper describes an architecture-centric self-adaptive approach and presents a simple application in a distributed system where it can be advantageous to switch architectures based on the workload being presented to the system. The self-adaptive framework is built on top of a generative system which comprises three software architectural alternatives, namely Single Thread (ST), Half-Sync/Half-Async (HS/HA) and Leaders-Followers (LFs). A software performance analysis tool called the Layered Queuing Network Solver (LQNS) is integrated into the framework to support the architecture selection process. A comparison of the performance of the three different software architecture alternatives is also presented. The results from this analysis are used to support the construction of a performance knowledge base and analysis policies for the self-adaptive system.

1 Introduction

The complexity of computer systems is increasing at a fast speed and the number of computing devices in use is growing dramatically (Parashar and Hariri, 2007). As a result, IT personnel have to face the burden of supporting tasks such as configuration, maintenance and system performance evaluation (Enterprise Managment Associates, 2006). Further, manual control of a distributed computing system or a Web system is prone to errors, time-consuming, and expensive. The goal of autonomic computing, initiated by IBM (Parashar and Hariri, 2007), is to define rules for a system for controlling its behavior so that the system regulates its actions to automatically configure, heal, protect, and optimize itself (Kephart and Chess, 2003). Many research projects related to autonomic computing have been started (Muller et al., 2006), but there is still a lack of research in the area of evaluating performance of software architectures supporting architecture-based self-adaption at runtime.

Software architectures have significant impact on the performance of a system, however, determining an optimal architecture early in the life cycle of a project is a challenging issue, For instance, from the concurrency management perspective, two efficient architectural alternatives have been proposed: Half-Sync/Half-Async (HS/HA) and Leader/Followers (LFs) (Schmidt et al., 2000). The question that an architect often has to address is that of finding an architecture