Chapter 8
Towards Autonomic Workload Management in DBMSs

Baoning Niu
Taiyuan University of Technology, China & Queen’s University, Canada

Patrick Martin
Queen’s University, Canada

Wendy Powley
Queen’s University, Canada

ABSTRACT
Workload management is the discipline of effectively managing, controlling and monitoring work flow across computing systems. It is an increasingly important requirement of database management systems (DBMSs) in view of the trends towards server consolidation and more diverse workloads. Workload management is necessary so the DBMS can be business-objective oriented, can provide efficient differentiated service at fine granularity and can maintain high utilization of resources with low management costs. We see that workload management is shifting from offline planning to online adaptation. In this paper we discuss the objectives of workload management in autonomic DBMSs and provide a framework for examining how current workload management mechanisms match up with these objectives. We then use the framework to study several mechanisms from both DBMS products and research efforts. We also propose directions for future work in the area of workload management for autonomic DBMSs.

INTRODUCTION
Workload management involves the monitoring and control of work entering a system. Its goal is to ensure that sufficient resources are allocated to a workload so that a business can meet its management objectives (IBM Corporation, 2003a).

DOI: 10.4018/978-1-60960-521-6.ch008
peting performance objectives. Third, Web-based applications, especially Web services (Erickson & Siau, 2008), introduce a need for flexible and guaranteed application service levels because they tend to involve unpredictable workloads, with a high rate of overall growth in workload size (D. H. Brown Associate, Inc, 2004). Allocating DBMS resources to competing workloads to meet performance objectives is a challenge.

We believe that workload management stands to benefit greatly from the new paradigm of autonomic computing. The goal of autonomic computing is to simplify system complexity by governing all computations within a given system (Mainsah, 2002). Autonomic computing systems have four fundamental characteristics (Ganek & Corbi, 2003): self-configuring, self-healing, self-optimizing and self-protecting. Self-configuring means systems can automatically adapt to dynamically changing environments. New features, applications and servers can be dynamically added to the enterprise infrastructure with no disruption of services. Self-healing is the ability of systems to discover, diagnose and react to disruptions. Such a system needs to be able to predict problems and take necessary actions to prevent the failures from impacting the services. For a system to be self-optimizing, it should monitor itself and tune resources automatically to maximize resource utilization to meet users’ performance requirements. Self-protecting means systems are able to anticipate, detect, identify, and protect themselves from attacks from anywhere.

The goal of this paper is to specifically examine the progress made towards providing autonomic workload management in DBMSs and, based on this examination, to identify directions for future research. Previous surveys of the autonomic computing area, such as those by Salehie and Tahvildari (2005) and Elnaifar, Powley, Benoit, and Martin (2003), tend to have a more general focus and use criteria focusing on the existence of facilities to support the general features of self-configuration, self-healing, self-optimization and self-protection.

We propose a framework, Autonomic Workload Management Framework (AWMF), for the examination of autonomic workload management and then analyze current products and research efforts with respect to it. The framework specifies a model of the key processes and functions in autonomic workload management as well as identifying criteria for evaluation.

The remainder of the paper is structured as follows. The following section discusses the objectives of workload management. We then outline our framework, AWMF, for autonomic workload management and examine workload management mechanisms taken from current DBMS products and research efforts and compare them using AWMF. A number of observations are derived from the comparison of the workload management mechanisms. We then validate AWMF using experiments with Query Scheduler, a prototype implementation of AWMF and finally summarize our survey and point out possible directions for future work.

**MANAGEMENT OBJECTIVES**

Workload management has evolved through three phases, from its infancy as a means for capacity planning (Lo & Douglas, 1986), to resource-oriented workload management, with the primary goal being resource utilization (D. H. Brown Associate, Inc, 2004; Castro, Tezulas, Yu, Berg, Kim, & Gfroerer, 2001), and finally to today’s performance-oriented workload management, with a focus on business objectives (IBM Corporation, 2003a). The objectives of workload management have remained the same, namely cost sharing and meeting Service Level Objectives (SLOs) (Menascé, 2004; IBM Corporation, 2003a), but its style has changed from offline analysis to online adaptation.

Cost sharing consolidates multiple applications onto a single server to improve resource utilization and cut down costs. An SLO objectively defines the
Related Content

Information Modeling Based on Semantic and Pragmatic Meaning
[www.igi-global.com/chapter/information-modeling-based-semantic-pragmatic/4375?camid=4v1a](www.igi-global.com/chapter/information-modeling-based-semantic-pragmatic/4375?camid=4v1a)

Full-Text Manipulation in Databases
[www.igi-global.com/chapter/full-text-manipulation-databases/20773?camid=4v1a](www.igi-global.com/chapter/full-text-manipulation-databases/20773?camid=4v1a)

Comparing Object-Oriented and Extended-Entity-Relationship Data Models
[www.igi-global.com/article/comparing-object-oriented-extended-entity/51151?camid=4v1a](www.igi-global.com/article/comparing-object-oriented-extended-entity/51151?camid=4v1a)

SQL Code Poisoning: The Most Prevalent Technique for Attacking Web Powered Databases
[www.igi-global.com/chapter/sql-code-poisoning/8025?camid=4v1a](www.igi-global.com/chapter/sql-code-poisoning/8025?camid=4v1a)