IDEA GROUP PUBLISHING **ITB13071**

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

This paper appears in the publication, *Advanced Topics in Database Research, vol. 5*
edited by Keng Siau © 2006, Idea Group Inc.

Chapter XII

Toward Autonomic DBMSs: A Self-Configuring Algorithm for DBMS Buffer Pools

Patrick Martin, Queen's University, Canada
Wendy Powley, Queen's University, Canada
Min Zheng, Queen's University, Canada

ABSTRACT

This chapter introduces autonomic computing as a means to automate the complex tuning, configuration, and optimization tasks that are currently the responsibility of the database administrator. We describe an algorithm called the dynamic reconfiguration algorithm (DRF) that can be implemented as part of an autonomic database management system (DBMS) to manage the DBMS buffer pools, which are a key resource in a DBMS. DRF is an iterative algorithm that uses greedy heuristics to find a reallocation that benefits a target transaction class. DRF uses the principle of goal-oriented resource management. We define and motivate the cost- estimate equations used in the algorithm and present the results of a set of experiments to investigate the performance of the algorithm.

INTRODUCTION

The explosion of the Internet and electronic commerce in recent years has database management system (DBMS) vendors scrambling to cope not only with the ever-increasing volumes of data to be managed, but also with the unique requirements precipitated by diverse data and unpredictable, often "bursty" access patterns. The addition of new features and functionality to DBMSs to address these issues has lead to increased system complexity. The management of DBMSs has traditionally been left to the experts — the database administrators — who monitor, analyze, and tweak the system for optimal performance. Given the increased complexity of DBMSs and the diverse and integrated environments in which they currently function, manual maintenance and tuning has become impractical, if not impossible.

DBMS parameter tuning is just one facet of tuning a database system, yet even this task has become a burden due to its complexity. Commercial database management systems typically provide upwards of 100 parameters that can be manually tuned. These parameters are often interconnected, so tuning one parameter may require an adjustment of one or more dependent resources. Determining optimal settings for tuning parameters requires knowledge of the characteristics of the system, the data, the workload, and of the interrelationships between them. These optimal settings often deteriorate over time, as the database characteristics change, or periodically, as the workload changes. With the varying and unpredictable patterns of electronic commerce workloads, changes in the workload tend to be more frequent and more extreme than those observed in traditional business environments. It is impractical for a database administrator to constantly monitor and tune the DBMS to adapt to these dynamic workloads. Instead, the system itself should be able to recognize or, where possible, predict workload changes, evaluate the benefit of reconfiguration, and independently take appropriate action.

Autonomic computing is an initiative spawned by IBM in 2001 to address the management problems associated with complex systems (Ganek & Corbi, 2003). IBM's use of the term "autonomic" is a direct analogy to the autonomic nervous system of the human body. The autonomic nervous system unconsciously regulates the body's low-level vital functions such as heart rate and breathing. The vision is to create computer systems that function much like the autonomic nervous system, that is, the low-level functionality and management of the system is attended to without conscious effort or human intervention. The goal is that system management will become the sole responsibility of the system itself.

In this chapter, we illustrate how the concept of autonomic computing can be applied to automate one aspect of DBMS tuning, that is, the tuning of the DBMS buffer pools. We present a self-tuning algorithm called the dynamic reconfiguration algorithm (DRF). This algorithm is based on the concept of goal-oriented resource management, which allows administrators to specify their expectations or goals for performance, while leaving it up to the system to decide how to achieve those goals (Nikolaou, Ferguson, & Constantopoulos, 1992). The database administrator (DBA) specifies average response-time goals for each transaction class in the workload. If one or more classes are not meeting their goals, then DRF chooses a reallocation of buffer pages to buffer pools that will improve the performance of the classes so that their goals can be met.