Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Decision Support Systems 38 (2005) 629–648
Decision Support Systems
www.elsevier.com/locate/dsw

Requirements-driven database systems benchmark method

Jia-Lang Seng[a,*], S. Bing Yao[b,1], Alan R. Hevner[c,2]

[a] *Deparment of MIS, National Chengchi University, Wenshan District, 116, Taipei, Taiwan, ROC*
[b] *XBD System Inc., Columbia, MD, USA*
[c] *Department of IS and DS, University of South Florida, Tampa, FL, USA*

Received 31 January 2003; accepted 30 June 2003
Available online 27 January 2004

Abstract

Benchmarks are the vital tools in the performance measurement, evaluation, and comparison of relational database management systems (RDBMS). Standard benchmarks such as the TP1, TPC-A, TPC-B, TPC-C, TPC-D, TPC-H, TPC-R, TPC-W, Wisconsin, and AS^{3}AP benchmarks have been used to assess the performance of relational database management systems. These benchmarks are synthetic and domain-specific. Test results from these benchmarks are estimates of possible system performance for certain pre-determined application types. Database system performance on actual database domain may vary significantly from those in the standard benchmarks. In this paper, we describe a new benchmark method that is computer-assisted and developed from the perspective of the user's requirements.

Keywords: Database benchmark; Performance measurement and evaluation; Relational database management systems; Requirements analysis; Workload characterization; Workload model; Synthetic benchmark

1. Introduction

A database benchmark is a standard set of executable instructions that are used to measure and compare the relative and quantitative performance of two or more database management systems through the execution of controlled experiments. Benchmark data such as throughput, jobs per time unit, and the inverse measure, response time, time per job unit, and other independent measures such as price performance ratio, equivalent database size, and Web interactions per second serve to predict and profile the system performance. It in turn assists a variety of user groups to make a decision on the system procurement, the capacity planning, and the bottleneck detection [11,26–28].

Database benchmarks comprise test databases and test workloads. They can be synthetic or empirical. Synthetic benchmarks emulate typical applications in a pre-determined problem domain and create a corresponding synthetic workload. Empirical benchmarks utilize real data and tests and re-invent the actual database applications. Though real workloads are ideal tests for systems, the costs of implementation of the actual environments usually outweigh the benefits provided by empirical benchmarks. Synthetic bench-

* Corresponding author. Tel.: +886-2-2938-7692.
E-mail addresses: jljan@nccu.edu.tw (J.-L. Seng), ahevner@coba.usf.edu (A.R. Hevner).
URL: http://www.coba.usf.edu/hevner.
[1] Tel.: 1-301-405-1111.
[2] Tel.: 1-813-974-6753.

0167-9236/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2003.06.002

marks are therefore the common approach chosen by vendors and users. However, current synthetic benchmarks are mostly domain pre-determined. Reproduction of test results is not guaranteed and variation can occur at the actual user settings. As Refs. [7,8,11,25] pointed out in their works that, although standard and synthetic benchmarks model typical applications, there is no proven way to assess the representative degree of the typical applications. Users must be aware that test results may not be reproduced in their environment. Users are advised to revise and re-execute these experiments. Standard benchmark results are rough estimates and only serve the purpose of relative comparison. Simulated workload varies and domain changes. The degree of variation depends on the approximation of the workloads in the real environments to those defined in the standard synthetic benchmarks. No interpolation of standard results is recommended. Refs. [10,12,18,29,30] also point out in their works that the use of pre-determined workloads is the root of the structural inadequacy of synthetic benchmarks. Domain dependency and application bound deters the further enhancement of synthetic benchmark. And, the structural deficiency causes the irreproduced and irrepresentative performance results. At present, standard benchmarks consist of:

- The TP1 benchmark [1,17], the TPC-A, and TPC-B benchmarks [19–21] are on-line transaction processing (OLTP) benchmarks simulating one bank transaction type, later retired by TPC in 1995.
- The TPC-C benchmark is a complex OLTP benchmark emulating order entry and inventory control transactions in a production environment to replace TPC-A and TPC-B benchmarks [15,16,21].
- The TPC-D benchmark, later revised and replaced by the TPC-H and TPC-R benchmarks, is a complex on-line analytic processing (OLAP) benchmark simulating decision support systems (DSS) transactions [22–24].
- The TPC-W benchmark is the latest TPC official benchmark and an electronic commerce (EC) benchmark to model the Internet bookstore transactional workloads [25].
- The Wisconsin benchmark is a relational query benchmark [2–6].
- The AS^3AP benchmark is a complex mixed workload benchmark [18,26–28].

These are standard relational database management systems (RDBMS) benchmarks. They use synthetic, domain-specific, and simulational workloads. They provide relative measurement and evaluation of system performance from pre-set profile. The key drawback is when the user domain differs from the standard domain and when the application workload deviates with the test workload, they cannot reflect the differences and changes in their design. Test results vary at user's domain and mislead user's decision.

In this research, we tackle the issue by proposing a domain-independent and application-independent synthetic workload model that is developed from the perspective of the user's requirements. This synthetic benchmark method differs from the current standard benchmarks in several aspects. First, it is a computer-assisted synthetic benchmark environment where the benchmark development is automated as much as we can. It is a benchmark to be perceived as a workload formulation process of requirements representation, transformation, and generation in an automated manner. We adopt the concept of requirements analysis to be the carrier to capture and compile user's requirements. The adoption is accomplished via the framework development from workload characterization. Data model, transaction model, and control model compose the carrier framework. Data analysis, transaction analysis, and control analysis apply the concept of requirements analysis. In specific, a high-level specification language, a translator of the language, and a set of generators are created to compose test databases and test transactions. User-friendly factor is the key concern to be a cost-effective design. Users enter their workload characteristics via commands, selects, and fill-ins. Translator automatically parses and produces the workload generation codes. Embedded SQL scripts and high-level programming statements are turned out. Test driver and result collector work with database generator and transaction generator to make test databases and issue database transactions against the system under test. In this study, we have conducted several benchmark experiments to test the method. We select the TPC-A, Wisconsin, AS^3AP, and TPC-C benchmarks as our standard test cases in the first set of experiments. We create a more generalized and extended case in the second set of experiments to show the generality of our method. Experimental results including the produced workload