Database Processing Benchmarks

Jérôme Darmont
Université de Lyon (Laboratoire ERIC), France

INTRODUCTION

Performance measurement tools are very important, both for designers and users of database systems, whether they are aimed at On-Line Transaction Processing (OLTP) or On-Line Analysis Processing (OLAP). Performance evaluation is useful to designers to determine elements of architecture, and more generally to validate or refute hypotheses regarding the actual behavior of a system. Thus, performance evaluation is an essential component in the development process of well-designed and scalable systems, which is nowadays of primary importance in the context of cloud computing. Moreover, users may also employ performance evaluation, either to compare the efficiency of different technologies before selecting a software solution or to tune a system.

Performance evaluation by experimentation on a real system is generally referred to as benchmarking. It consists in performing a series of tests on a given system to estimate its performance in a given setting. Typically, a database benchmark is constituted of two main elements: a data model (conceptual schema and extension) and a workload model (set of read and write operations) to apply on this dataset, with respect to a predefined protocol. Most benchmarks also include a set of simple or composite performance metrics such as response time, throughput, number of input/output, disk or memory usage, etc.

The aim of this article is to present an overview of the major families of state-of-the-art data processing benchmarks, namely transaction processing benchmarks and decision support benchmarks. We also address the newer trends in cloud benchmarking. Finally, we discuss the issues, tradeoffs and future trends for data processing benchmarks.

BACKGROUND

Transaction Processing Benchmarks

In the world of relational database benchmarking, the Transaction Processing Performance Council (TPC) plays a preponderant role. The mission of this non-profit organization is to issue standard benchmarks, to verify their correct application by users, and to regularly publish the results of performance tests. Classical TPC benchmarks all share variants of a classical business database (customer-order-product-supplier) and are only parameterized by a scale factor that determines the database size (e.g., from 1 GB to 100 TB).

The TPC benchmark for transactional databases, TPC-C (TPC, 2010a), has been in use since 1992. It is specifically dedicated to OLTP applications, and features a complex database (nine types of tables bearing various structures and sizes), and a workload of diversely complex transactions that are executed concurrently. The performance metric in TPC-C is throughput, in terms of transactions. TPC-C was complemented in 2007 by TPC-E (TPC, 2010b), which simulates a brokerage firm with the aim of being representative of more modern OLTP systems that those modeled in TPC-C. TPC-E's principles and features are otherwise very similar to TPC-C's.

There are currently very few alternatives to TPC-C and TPC-E, although some benchmarks have been proposed to suit niches in which there is no standard benchmark. For instance, OO7 (Carey et al., 1993) and OCB (Darmont & Schneider, 2000) are object-oriented database benchmarks modeling engineering applications (e.g., computer-aided design, software engineering). However, their complexity makes both these benchmarks hard to understand and implement. Moreover, with objects in databases being more com-

DOI: 10.4018/978-1-4666-5888-2.ch167

monly managed in object-relational systems nowadays, object-relational benchmarks such as BUCKY (Carey et al., 1997) and BORD (Lee et al., 2000) now seem more relevant. Such benchmarks focus on queries implying object identifiers, inheritance, joins, class and object references, multivalued attributes, query unnesting, object methods, and abstract data types. However, typical object navigation is considered already addressed by object-oriented benchmarks and is not taken into account. Moreover, object-relational database benchmarks have not evolved since the early 2000's, whereas object-relational database systems have.

Finally, XML benchmarks aim at comparing the various XML storage and querying solutions developed since the late nineties. From the early so-called XML application benchmarks that implement a mixed XML database that is either data-oriented (structured data) or document-oriented (in general, random texts built from a dictionary), XBench (Yao et al., 2004) stands out. XBench is indeed the only benchmark proposing a true mixed dataset (i.e., data *and* document-oriented) and helping evaluate all the functionalities offered by XQuery. FlexBench (Vranec & Mlýnková, 2009) also tests a large set of data characteristics, but also proposes query templates that allow modeling multiple types of applications. Finally, Schmidt et al. (2009) and Zhang et al. (2011) propose benchmarks that are specifically tailored for testing logical XML model-based systems, namely native XML and XML-relational database management systems, respectively.

Decision Support Benchmarks

Since decision-support benchmarks are currently a de facto subclass of relational benchmarks, the TPC again plays a central role in their standardization. TPC-H (TPC, 2013) has long been the only standard decision-support benchmark. It exploits a classical product-order-supplier database schema, as well as a workload that is constituted of twenty-two SQL-92, parameterized, decision-support queries and two refreshing functions that insert tuples into and delete tuples from the database. Query parameters are randomly instantiated following a uniform law. Three primary metrics describe performance in terms of power, throughput, and a combination of power and throughput.

However, TPC-H's database schema is not a star-like schema that is typical in data warehouses. Furthermore, its workload does not include any OLAP query. TPC-DS (TPC, 2012) now fills in this gap. Its schema represents the decision-support functions of a retailer under the form of a constellation schema with several fact tables and shared dimensions. TPC-DS' workload is constituted of four classes of queries: reporting queries, ad-hoc decision-support queries, interactive OLAP queries, and extraction queries. SQL-99 query templates help randomly generate a set of about five hundred queries, following non-uniform distributions. One primary throughput metric is proposed in TPC-DS, which takes both query execution and the data warehouse maintenance into account.

There are, again, few decision-support benchmarks out of the TPC, but with TPC-DS having been under development for almost eight years, alternative data warehouse benchmarks were proposed. Published by the OLAP council, a now inactive organization founded by OLAP vendors, APB-1 (OLAP Council, 1998) was the first of them and actually predates TPC-DS. APB-1 has been intensively used in the late nineties. However, APB-1 is very simple and rapidly proved limited to evaluate the specificities of various activities and functions.

Thus, more elaborate alternatives were proposed, such as DWEB (Darmont et al., 2007), which can be parameterized to generate various ad-hoc synthetic data warehouses and workloads that include typical OLAP queries, and SSB (O'Neil et al., 2009), which is based on TPC-H's database remodeled as a star schema and features a query workload that provides both functional and selectivity coverage.

Eventually, benchmarks also fill in niches that are not covered by the TPC. As SSB, XWeB (Mahboubi & Darmont, 2010) derives from TPC-H, but proposes of a test data warehouse based on a unified reference model for XML warehouses and its associate XQuery decision-support workload. RTDW-bench (Jedrzejczak et al., 2012) is also based on TPC-H. It is designed for testing the ability of a real-time data warehouse to handle a transaction stream without delay, given an arrival rate. Bär and Golab (2012) also propose a benchmark for stream data warehouses that measures the freshness of materialized views. Finally, a couple of benchmarks are even more specific (and unrelated from TPC-H), e.g., Spadawan (Lopes Siqueira et al., 2010), which allows performance evaluation of spe-