Data-Centric Benchmarking

Jérôme Darmont
Université de Lyon, Lyon 2, ERIC EA3083, France

INTRODUCTION

In data management, both system designers and users casually resort to performance evaluation. On one hand, designers need to test architectural features and hypotheses regarding the actual (vs. theoretical) behavior of a system, especially in terms of response and scalability. Performance tuning also necessitates accurate performance evaluation. On the other hand, users are also keen on comparing the efficiency of different technologies before selecting a software solution. Thence, performance measurement tools are of premium importance in the data management domain.

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 data-centric 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. Both models may be parameterized. Most benchmarks also include a set of simple or composite performance metrics such as response time, throughput, number of input/output operations, disk or memory usage, etc.

The Transaction Processing Performance Council (TPC), a non-profit organization founded in 1988, plays a preponderant role in data-centric benchmarking. Its mission is to issue standard benchmarks, to verify their correct application by the industry, and to publish performance test results. TPC members include all the major industrial actors from the database field.

The aim of this chapter is to present an overview of the major past and present state-of-the-art data-centric benchmarks. Our review includes the TPC standard benchmarks, but also alternative or more specialized benchmarks. We survey benchmarks from three families: transaction benchmarks aimed at On-Line Transaction Processing (OLTP), decision-support benchmarks aimed at On-Line Analysis Processing (OLAP) and big data benchmarks. Eventually, we discuss the issues, tradeoffs and future trends in data-centric benchmarking.

DOI: 10.4018/978-1-5225-2255-3.ch154

BACKGROUND

Transaction Processing Benchmarks

The first TPC benchmark for relational, transactional databases, TPC-C (TPC, 2010), has been in use since 1992. TPC-C features a complex business database (a classical customer-order-product-supplier model with 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 transaction throughput. As all TPC benchmarks, TPC-C's only parameter is a scale factor SF that determines data size. TPC-C was complemented in 2007 by TPC-E (TPC, 2015a), which simulates a brokerage firm with the aim of being representative of more modern OLTP systems. In its principles and features, TPC-E is otherwise very similar to TPC-C.

There are few alternatives to TPC-C and TPC-E for relational applications. Yet, some benchmarks fit niches where 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 or software engineering. However, their complexity makes both these benchmarks hard to understand and implement. Moreover, with objects in databases being more commonly 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.

Similarly, 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 and 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

TPC-H (TPC, 2014a) 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 multidimensional schema that is typical in data warehouses. Furthermore, its workload does not include any true OLAP query. TPC-DS (TPC, 2015b) 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. TPC-DS features one primary throughput metric that takes both query execution and data warehouse maintenance into account.

Given the primordial importance of data integration in many data-centric (including data warehousing) scenarios, TPC-H was recently complemented by TPC-DI (TPC, 2014b). TPC-DI focuses on Extract, Load and Transform (ETL) processes. Data are first generated in a staging area as if they were extracted from a virtual retail brokerage firm's operational databases. Then, data are transformed through, e.g., type conversions, attribute splits or merges, and error checks. Finally, data are loaded into a warehouse constituted of five fact tables and eight dimension tables. There are two load phases: an initial, so-called historical load, and then incremental updates. Transformations are different in these two phases. TPC-DI's main metric is a combination of throughputs from the historical load and two incremental updates.

There are, again, few decision-support benchmarks out of the TPC, but with TPC-DS having had an eight-year long development, alternative data warehouse benchmarks were proposed. Published by the OLAP council, a now inactive