# The mixed workload CH-benCHmark

Dagstuhl "Robust Query Processing"
Breakout Group "Workload Management"[1]

## ABSTRACT

While standardized and widely used benchmarks address either operational or real-time Business Intelligence (BI) workloads, the lack of a hybrid benchmark led us to the definition of a new, complex, mixed workload benchmark, called mixed workload CH-benCHmark. This benchmark bridges the gap between the established single-workload suites of TPC-C for OLTP and TPC-H for OLAP, and executes a complex mixed workload: a transactional workload based on the order entry processing of TPC-C **and** a corresponding TPC-H-equivalent OLAP query suite run **in parallel** on the **same** tables in a **single** database system. As it is derived from these two most widely used TPC benchmarks, the CH-benCHmark produces results highly relevant to both hybrid and classic single-workload systems.

## Categories and Subject Descriptors

C.4 [**Performance of Systems**]

## General Terms

Measurement, Performance, Standardization

## 1. INTRODUCTION

Traditionally, database workloads are categorized as strictly Online Transaction Processing (OLTP), strictly Online Analytical Processing (OLAP), or a hybrid mix of the two. OLTP systems service transaction-oriented applications vital to the day-to-day operations of a company, and thus focus on providing high availability and low latency to a large number of users. OLAP systems analyze large amounts of business data to support strategic decision making, e. g., computing sales revenue of a company by products across regions and time, or fraud detection, and thus tend to support fast scans and complex data algorithms for a relatively small number of users.

The Transaction Processing Performance Council (TPC) offers OLTP and Decision Support (DS) benchmarks recognized by the industry for almost 20 years as the standard to measure system wide OLTP and DS performance. However, recently demand has emerged for data warehouses that support complex mixed workloads comprised of a richer variety of queries than the traditional OLTP vs. BI. For example, Gartner characterizes complex mixed workloads as the invocation of analytics and BI-oriented functionality by OLTP applications and describe complex mixed workloads as consisting of four components: continuous data loading, batch (expected) data loading, large numbers (thousands per day) of standard reports, and random, unpredictable, ad-hoc query users [3].

Gartner observes that clients report that performance is problematic when processing complex mixed workloads. Our own experiences second this observation; at the September 2010 Dagstuhl Seminar on Robust Query Processing, a subgroup composed of the authors of this paper focused on the role of workload management in robust query processing. We all saw a need, currently unmet by existing TPC benchmarks, for a benchmark that measures performance with regard to a complex mixed database workload.

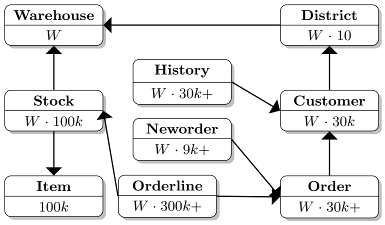
We describe here the CH-benCHmark, which models these components and which we intend as a step towards modeling complex mixed workloads. We particularly note the work of [1, 2], who propose the Composite Benchmark for Transaction processing and operational Reporting (CBTR), which includes OLTP and reporting components. The benchmark we propose models three of the four components described by Gartner: continuous data loading, batch data loading, and large numbers of standard reports. As discussed below, we model ad-hoc queries only to the extent that existing TPC benchmarks model them.

The remainder of the paper is organized as follows: Section 2 gives a brief overview of TPC-C and TPC-H. Section 3 defines the CH-benCHmark. Section 4 describes some metrics for interpreting results of our benchmark. Section 5 shows some preliminary results running the CH-benCHmark on a reference platform. Section 6 identifies open challenges. We conclude with Section 7.

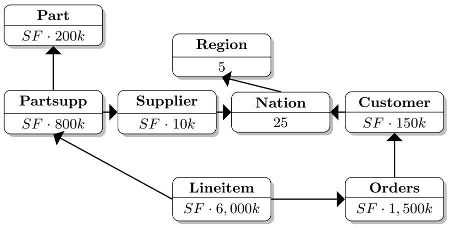
[1]Richard Cole (ParAccel), Florian Funke (TU München), Leo Giakoumakis (Microsoft), Wey Guy (Microsoft), Alfons Kemper (TU München), Stefan Krompass (TU München), Harumi Kuno (HP Labs), Raghunath Nambiar (Cisco), Thomas Neumann (TU München), Meikel Poess (Oracle), Kai-Uwe Sattler (TU Ilmenau), Michael Seibold (TU München), Eric Simon (SAP), Florian Waas (Greenplum)

[2]Over time the terms OLAP, DS have been amalgamated into the broader term Business Intelligence (BI). In this paper we will refer to OLAP, DS, and BI as necessary.


*DBTest'11,* 06-JUN-2011, Athens, Greece




(a) TPC-C




(b) TPC-H

**Figure 1: TPC-C and TPC-H schemas**

## 2. OVERVIEW OF THE TPC AND ITS MAJOR BENCHMARKS: TPC-C AND TPC-H

The Transaction Processing Performance Council (TPC) is a non-profit corporation founded to define vendor-neutral transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. Unlike other benchmark standardization organizations, such as SPEC and SPC, TPC's benchmarks measure the performance of large, complex systems involving hundreds of processors, sometimes in multi-tier architectures, and thousands of disks. Their results, published on the TPC website, are meticulously documented and verified by TPC certified and vendor independent auditors. TPC's most popular benchmarks, judged by their tenure and body of publications, are TPC-C for OLTP and TPC-H for DS applications.

The TPC has defined TPC-E, an OLTP benchmark specification that includes some more recent aspects of OLTP systems, such as a mandatory minimal redundancy level to guarantee access to data on durable media in case of a single durable media failure and mandatory reporting of recovery time. However, the business models of TPC-H and TPC-E are different. TPC-H models a retailer while TPC-E models a brokerage firm. The inherent differences in their database schemas and data itself prohibit an integration of the two benchmarks. This is the main reason for choosing TPC-C over TPC-E.

### 2.1 TPC-C

TPC Benchmark C (TPC-C) [6] models an OLTP workload. Since its establishment in 1992 it has undergone several revisions to keep up with technological changes. During its tenure of now almost 20 years it has been accepted in the industry as the most credible transaction processing benchmark, with published results across all major hardware and database platforms. Modeled after actual production applications and environments, it evaluates key performance factors such as user interface, communications, disk I/Os, data storage, and backup and recovery using a mixture of read-only and update-intensive business transactions: New-Order, Payment, Order-Status, Delivery, and Stock-Level. See [6] for more information about these transactions.

The TPC-C schema, depicted in Figure 1(a), contains nine tables: Warehouse, Stock, Item, History, New-Order, Orderline, District, Customer, and Order. TPC-C performance is defined as the Maximum Qualified Throughput (MQTh) rating of the system, expressed in tpmC. Using a well defined mix of the above transactions, tpmC is the number of orders a system can process per minute. Throughput is driven by the activity of the emulated terminals connected to each modeled warehouse. To increase throughput, more warehouses and their associated terminals must be configured. Each warehouse requires a number of rows to populate the database along with some storage space to maintain the data generated during a defined period of activity called a 60-day period. These requirements define how storage space and database population scale with throughput. The scaling requirements maintain the ratio between the transaction load presented to the system under test (SUT), the cardinality of the tables accessed by the transactions, the required space for storage, and the number of terminals generating the transaction load. For each active warehouse in the database, the SUT accepts requests for transactions from a population of 10 terminals. The Warehouse table is used as the base unit of scaling. The cardinality of all other tables (except for Item) is a function of the number of configured warehouses (cardinality of the Warehouse table). This number, in turn, determines the load applied to the SUT which results in a reported throughput.

TPC-C performance reported in a benchmark publication is the transaction throughput of new orders during steady state condition (tpmC). Performance is measured during the measurement interval, which must begin after the system reaches steady state, be long enough to generate reproducible throughput results representative of performance that would be achieved during a sustained eight hour period, and extend uninterrupted for a minimum of 120 minutes.

### 2.2 TPC-H

First established in 1999, TPC benchmark H (TPC-H) [5, 7] is a decision support benchmark, which models decision support systems that examine large volumes of data, execute queries with a high degree of complexity, and give answers to critical business questions. It consists of a suite of business oriented ad-hoc queries and concurrent data modifications, i. e., insert and delete statements. The queries and data populating the database model a wholesale supplier that must manage sell, or distribute products worldwide.

The TPC-H schema, depicted in Figure 1(b) consists of eight tables, which are populated prior to the performance test according to a scale factor (SF). The scale factor determines the size of the raw data, not their representation inside a database. The two largest tables, Lineitem and Orders contain about 83 percent of the data. The size of all tables, except for nation and region scales proportionally with the scale factor.

The performance test consists of two phases. The first phase is a single-user run, which, in single user mode, starts by inserting rows into Lineitem and Orders, continues executing 22 queries and finishes by deleting rows from Lineitem and Orders. The second phase is a multi-user run, which runs multiple single-user runs simultaneously. The single-user run stresses a system's ability to parallelize operations such that the answer for a given request can be obtained in the least amount of time, as desired for overnight batch job processing. The multi-user run stresses the system's ability to schedule concurrent requests from multiple users to increase overall system throughput. TPC-H performance is measured in the so-called composite performance metric, expressed in queries completed per hour (QphH). It equally weights the contribution of the single-user and the multi-user runs by calculating the geometric mean of the single- and multi-user performances.

With $SF$ being the scale factor, $T(q_i)$ the elapsed time of Query $i$ and $T(u_i)$ the elapsed time of Update Operation $i$, $S$ the number of emulated users and $T_s$ the elapsed time of the multi user run, the single user performance $P_{\text{singleUser}}$ and multi-user performances $P_{\text{multiUser}}$ are defined as follows:

$$P_{\text{singleUser}} = \frac{3600 * SF}{\sqrt[24]{\Pi_{i=1}^{22} T(q_i) * \Pi_{i=1}^{2} T(u_i)}}$$

$$P_{\text{multiUser}} = \frac{22 * S * 3600 * SF}{T_S}$$

$$P_{\text{composite}} = \sqrt{P_{\text{singleUser}} * P_{\text{multiUser}}}$$

| | Relation | CH-benCHmark scaling | Original TPC-H scaling |
|---|---|---|---|
| accessed by OLTP transactions only | WAREHOUSE | | |
| | DISTRICT | | |
| | HISTORY* | | |
| accessed by OLTP transactions **and** OLAP queries | NEW-ORDER* | $9 \times w$ | n.a. |
| | STOCK | $10 \times w$ | n.a. |
| | CUSTOMER | $3 \times w$ | 15 |
| | ORDERLINE* (LINEITEM) | $300 \times w$ | 600 |
| | ORDERS* | $30 \times w$ | 150 |
| | ITEM (PART) | $100 \times w$ | 20 |
| accessed by OLAP queries only | SUPPLIER | 1 | 1 |
| | NATION | fixed | fixed |
| | REGION | fixed | fixed |

**Table 1: The tables in the CH-benCHmark. The scaling of the tables in the CH-benCHmark and TPC-H differs. The size (and thus the scaling) of the tables marked with an asterisk change over the course of a benchmark run.**

# 3. DESIGN OF THE CH-benCHmark

The CH-benCHmark combines the unmodified TPC-C schema and transactions and an adapted version of the TPC-H queries to form a complex mixed workload like that described by Gartner [3]. The TPC-C New-Order and Payment transactions represent continuous data loading, the Delivery transaction represents batch data loading, and Order-Status and Stock-Level represent standard reports. The TPC-H queries correspond to both standard report and tactical business analytics queries.

## 3.1 The Schema of the CH-benCHmark

As the schemas of both benchmarks (cf. Figure 1) model businesses which "must manage, sell, or distribute products or services" [6, 7], they share some similarities. The relations ORDER(S) and CUSTOMER exist in both schemas, and both ORDER-LINE (TPC-C) and LINEITEM (TPC-H) model entities that are sub-entities of ORDER(S). However, they differ in other aspects, such as their scaling models.

The CH-benCHmark keeps all TPC-C entities and relationships completely unmodified and integrates the likewise unchanged relations SUPPLIER and REGION from the TPC-H schema. We slightly modify the NATION relation as described below. These relations are frequently used in TPC-H queries and allow a non-intrusive integration into the TPC-C schema.

TPC-C and TPC-H follow different scaling models. TPC-C follows a continuous scaling model, where rows in the warehouses table are increased with system performance. Contrary, TPC-H follows the fixed scale factor model, where the database size is set by a scale factor regardless of system performance. We adapt the scaling model of TPC-H to that of TPC-C. That is, WAREHOUSE, STOCK, ITEM, HISTORY, NEWORDER, ORDERLINE, DISTRICT, CUSTOMER, and ORDER scale according to the TPC-C rules. The relation SUPPLIER is populated with a fixed number (10,000) of entries. Thereby, an entry in STOCK can be uniquely associated with its SUPPLIER through the relationship STOCK.S_I_ID $\times$ STOCK.S_W_ID $mod$ 10,000 = SUPPLIER.SU_SUPPKEY. A CUSTOMER's NATION is identified by the first character of the field C_STATE. TPC-C specifies that this first character can have 62 different values (upper-case letters, lower-case letters and numbers), therefore we chose 62 nations to populate NATION (TPC-H specifies 25 nations). The primary key N_NATIONKEY is an identifier according to the TPC-H specification. Its values are chosen such that their associated ASCII value is a letter or number (i. e., N_NATIONKEY $\in [48, 57] \cup [65, 90] \cup [97, 122]$). Therefore no additional calculations are required to skip over the gaps in the ASCII code between numbers, upper-case letters and lower-case letters. REGION contains the five regions of these nations. Relationships between the new relations are modeled with simple foreign key fields (NATION.N_REGIONKEY and SUPPLIER.SU_NATIONKEY).

## 3.2 Scaling of the CH-benCHmark

Table 1 shows how TPC-H relations scale in the CH-benCHmark schema. The first column lists all relations as they are used in the CH-benCHmark schema. Where CH-benCHmark relation names differ from the ones in TPC-H, the name of the original TPC-H relation is given in parenthesis. For the tables that are accessed by the OLAP queries in our CH-benCHmark, columns 2 and 3 compare how the tables scale in the CH-benCHmark and in the TPC-H benchmark. The scaling is expressed relative to the SUPPLIER table (10,000 entries). The scaling of the various relations is very different from the original TPC-H scaling. Also note that, because we keep the tuples in the SUPPLIER relation constant, the scaling of relations ORDERS, ORDERLINE, HISTORY, and NEW-ORDER change relative to the SUPPLIER relation during the course of a benchmark run (OLTP transactions insert tuples into these relations). There are no tuples added to CUSTOMER, STOCK, and ITEM. Consequently, the ratio of the size of these tables relative to SUPPLIER do not change. The differences in scaling have a bearing on the execution of OLAP queries in the CH-benCHmark compared to their TPC-H counterparts. Consequently, the OLAP performance in our CH-benCHmark cannot easily be inferred from the performance of a similarly-sized TPC-H installation.

## 3.3 Transactions and Queries

As illustrated in the overview in Figure 2, the workload consists of the five TPC-C transactions and 22 queries, adopted and adapted from TPC-H. Since the TPC-C schema is

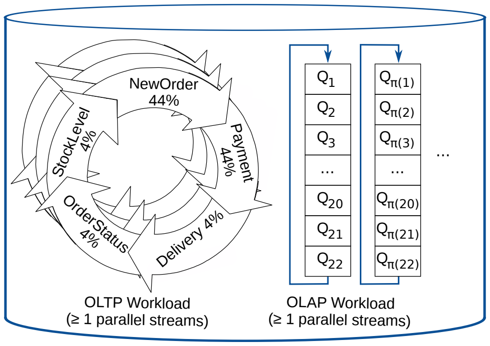



**Figure 2: Benchmark overview: OLTP and OLAP on the same data**

an unmodified subset of the CH-benCHmark schema, the original transactions can be executed without modification:

**New-Order:** This transaction enters an order with multiple order-lines into the database. For each order-line, 99% of the time the supplying warehouse is the home warehouse. The home warehouse is a fixed warehouse ID associated with a terminal. To simulate user data entry errors, 1% of the transactions fail and trigger a roll-back.

**Payment:** A payment updates the balance information of a customer. 15% of the time, a customer is selected from a random remote warehouse, in the remaining 85%, the customer is associated with the home warehouse. The customer is selected by last name in 60% of the cases and else by his three-component key.

**Order-Status:** This read-only transaction is reporting the status of a customer's last order. The customer is selected by last name 60% of the time. If not selected by last name, he is selected by his ID. The selected customer is always associated with the home warehouse.

**Delivery:** This transaction delivers 10 orders in one batch. All orders are associated with the home warehouse.

**Stock-Level:** This read-only transaction operates on the home warehouse only and returns the number of those stock items that were recently sold and have a stock level lower than a threshold value.

The distribution over the five transaction types conforms to the TPC-C specification (cf. Figure 2), resulting in frequent execution of the New-Order and Payment transactions. The CH-benCHmark deviates from the underlying TPC-C benchmark by not simulating the terminals and by generating client requests without any think-time, as proposed by [8]. Since the transactions themselves remain the same as in TPC-C, the CH-benCHmark results are directly comparable to existing TPC-C results with the same modifications, e. g., VoltDB [8]. Moreover, these changes can be easily applied to existing TPC-C implementations in order to produce CH-benCHmark-compatible results.

For the OLAP portion of the workload, we adapt the 22 queries from TPC-H to the CH-benCHmark schema[1]. In reformulating the queries to match the slightly different schema, we made sure that their business semantics and syntactical structure were preserved. For example, query 5 lists the revenue achieved through local suppliers (cf. Listing 1 and 2). Both queries join similar relations, have similar selection criteria, perform summation, grouping and sorting. The original query 5 joins six relations, while the modified joins seven relations. Another difference is that the original performs more arithmetic operations on each tuple, SUM(L_EXTENDEDPRICE*(1 - L_DISCOUNT)), while the modified query performs a simple aggregation (SUM(OL_AMOUNT)).

Contrary to TPC-H, the CH-benCHmark does not require refresh functions because the TPC-C transactions are continuously updating the database. The following section particularizes when queries have to incorporate these updates.

### 3.4 Benchmark Parameters

The CH-benCHmark has four scales: First, the database size is variable. As in TPC-C, the size of the database is specified through the number of warehouses, as explained in Section 3.2. The second scale is the composition of the workload. It can be comprised of analytical queries only, transactions only or any combination of the two. The workload mix is specified as the number of parallel OLTP and DS streams that are connected to the database system. An OLTP stream dispatches random TPC-C transactions sequentially with the distribution described in the official specification [6]. A DS stream performs continuous iterations over the query set which is comprised of all 22 queries. As depicted in Figure 2, each stream starts with a different query to avoid caching effects. The third scale is the isolation level. Lower levels like read committed allow for faster processing, while higher levels guarantee higher quality results for both transactions and queries. The final scale is the freshness of the data, which only applies if the workload mix contains both OLTP and OLAP components. Data freshness is specified as the time or the number of transactions after which newly issued queries have to incorporate the most recent data. This allows for database architectures that have a single data set for both workloads and those that devise a delta to run the benchmark.

## 4. INTERPRETING THE RESULTS OF THE BENCHMARK

OLAP and OLTP benchmarks differ in terms of query characteristics (short transactions vs. complex analytical queries), scaling (number of warehouses vs. scaling factor of fact table) as well as the overall metrics. Multiple performance indicators are thus required to compare systems or system configurations, depending on the primary use of the system and the goal of the benchmarking.

An obvious first metric is to calculate a weighted average or a harmonic mean of the TPC-C and the TPC-H performance metrics. However, this requires a normalization of these two metrics which is difficult due to their different domains and meanings.

Depending on the primary usage of the system, one of the two workloads can be seen as a disturbance of the other (primary) workload. In combination with performance for the part of the workload that is the primary use case and provided that the run meets the specified requirements such as freshness and isolation level, we can use the ratio between tpmC and QphH as a metric. Assuming that the transactional part is the primary use case for the benchmark, $\frac{tpmC}{QphH}@tpmC$ as a measure for gauging which part of the benchmark suffers more from the parallel execution given $n$ OLTP and $m$ OLAP query streams (similarly, we can declare $\frac{tpmC}{QphH}@QphH$ for workloads where the analyti-

[1]The SQL for all 22 queries can be found at `http://www-db.in.tum.de/research/projects/CH-benCHmark/`.

```sql
SELECT n_name, SUM(ol_amount) AS revenue
FROM customer, "order", orderline, stock, supplier,
  nation, region
WHERE c_id=o_c_id AND c_w_id=o_w_id AND c_d_id=o_d_id
  AND ol_o_id=o_id AND ol_w_id=o_w_id
  AND ol_d_id=o_d_id
  AND ol_w_id=s_w_id AND ol_i_id=s_i_id
  AND mod((s_w_id * s_i_id),10000)=su_suppkey
  AND ascii(SUBSTRING(c_state, 1, 1))=su_nationkey
  AND su_nationkey=n_nationkey
  AND n_regionkey=r_regionkey
  AND r_name='[REGION]' AND o_entry_d>='[DATE]'
GROUP BY n_name ORDER BY revenue DESC
```

**Listing 1: CH-benCHmark query 5**

```sql
SELECT n_name, SUM(l_extendedprice * (1 - l_discount)
  ) AS revenue
FROM customer, orders, lineitem, supplier, nation,
  region
WHERE c_custkey = o_custkey
  AND l_orderkey = o_orderkey
  AND l_suppkey = s_suppkey
  AND c_nationkey = s_nationkey
  AND s_nationkey = n_nationkey
  AND n_regionkey = r_regionkey
  AND r_name = '[REGION]'
  AND o_orderdate >= DATE '[DATE]'
  AND o_orderdate < DATE '[DATE]' + INTERVAL '1' YEAR
GROUP BY n_name ORDER BY revenue DESC
```

**Listing 2: TPC-H Query 5**

cal queries dominate). For this comparison, we first compute the ratio between the tpmC and QphH measurements from the runs where $n$ OLTP and $m$ OLAP streams run in isolation. We then compare this ratio to the ratio derived from running the mixed workload with $n$ OLTP and $m$ OLAP streams executed in parallel. A higher ratio for the parallel execution than for the case where the workload parts run in isolation, means that the database system provides better service for OLTP transactions in the parallel execution.

Based on the previous case, a third metric models robustness by comparing performance degradation of the primary workload to the homogeneous case. This could be further refined by introducing service classes (e. g., a subset of analytical queries and certain update transactions) instead of considering only two workloads.

Finally, the configuration of a system to achieve best performance for both the analytical and the transactional part of the workload as well as the freshness and isolation level parameters can be interpreted as a multi-objective optimization problem. In this way, different systems or system configurations can be compared in terms of the Pareto optimum: system $A$ is better than system $B$ if it dominates $B$ in the multi-dimensional space, i. e., if it is as good as or better in all dimensions and better in at least one dimension.

## 5. FIRST RESULTS FROM POSTGRESQL

We conducted our experiments on a commodity server with two Intel X5570 Quad-Core-CPUs with 8MB cache each and 64GB RAM. The machine had 16 2.5" SAS disks with 300GB that were configured as RAID 5 with two logical devices. As operating system, we were using an Enterprise-grade Linux running a Linux kernel v2.6 that was shipped with the distribution. The clients that submitted the OLTP transactions (TPC-C) and the OLAP queries (TPC-H) were executed on the same machine.

We implemented the CH-benCHmark based on PostgreSQL 9.0.3. We loaded 12 warehouses into the database, resulting in a disk space consumption for the tables and indexes of about 1054MB and 190MB, respectively. We configured PostgreSQL to use "read committed" as isolation level. The data and the transaction log are stored separately on the two logical volumes. We chose the memory configuration for the database such that eventually the data is memory-resident. We note that the experiments are not meant to be a thorough evaluation of PostgreSQL but a demonstration of the variability of the benchmark and the reporting.

We ran experiments varying the number of OLTP and OLAP query streams, similar to [4], where the CH-benCHmark was applied to main memory database systems. Each experiment consists of a 5 minute ramp-up phase and a 20 minute measurement phase. We chose the length of the ramp-up phase such that the system is in steady state before the measurements begin. Our first set of experiments was "OLTP-only", and varied the number of OLTP streams while setting the number of OLAP query streams to 0. Figure 3 plots throughput numbers, measured in completed NewOrder transactions per minute ($tpmC$), and shows the characteristic load-performance curve. While the system is in underload, performance (throughput) increases with increasing load (number of streams) until performance hits a maximum throughput of about 42000 tpmC at 64 streams, i. e., the system is saturated. After saturation an increasing number of streams decreases throughput because the system is driven in overload.

Our second set of experiments was to run the OLAP-only case with 1 and eight OLAP query streams, respectively, and the number of OLTP streams set to 0. We conducted the experiments with a newly loaded 12 warehouse database. Table 2 shows average response times of the queries completed in the measurement interval, the geometric mean of average response times, the duration to complete a group of 22 queries, and the number of completed queries per hour (QphH). As can be seen, the average response time does not change significantly as we increase the number of OLAP query streams from one to eight. Since there is no resource contention (data fits in main memory, number of streams $\leq$ number of CPU cores, maximum memory bandwidth not reached, no conflicting locks) performance does not degrade significantly. Consequently, throughput increases from 895.6 QphH to 8×863.4=6907.2 QphH.

The last set of experiments executed the mixed workload with varying numbers of OLTP and OLAP query streams. With one OLTP and one OLAP query stream, throughput of the OLTP stream is similar to the OLTP-only experiment (about 5200 tpmC). Assuming that the transactional part is the primary use case for the experiments, we can use the tpmC measurements and the ratio introduced in Section 4 to gauge the impact of the parallel execution on OLTP and OLAP. For this case, the metric for the runs in isolation is $5.7@5084tpmC$ and for the parallel execution $6.5@5188tpmC$, confirming that the database system processes the two streams concurrently without performance degradation. The situation changes with eight OLTP streams and a single OLAP query stream. Comparing the OLTP throughput numbers for the OLTP-only case (11875 tpmC) and the mixed workload (12550 tpmC), we see that performance is comparable because the system is in underload and resource contention is low. However, comparing $13.3@11875tpmC$ (OLTP and OLAP run in isolation) with $20.0@12550tpmC$ (mixed workload) reveals that the OLAP

| Q# | 1 query stream single threaded OLTP: OLTP throughput | 1 query stream single threaded OLTP: response times (ms) | 1 query stream 8 OLTP streams: OLTP throughput | 1 query stream 8 OLTP streams: response times (ms) | 4 query streams 16 OLTP streams: OLTP throughput | 4 query streams 16 OLTP streams: response times (ms) | no OLTP: 1 Q. stream response times (ms) | no OLTP: 8 Q. streams response times (ms) |
|---|---|---|---|---|---|---|---|---|
| Q1 | new order: 5188 *tpmC* | 5203 | new order: 12550 *tpmC* | 6502 | new order: 18356 *tpmC* | 15214 | 4466 | 4346 |
| Q2 | | 1068 | | 1017 | | 2009 | 986 | 964 |
| Q3 | | 187 | | 243 | | 470 | 166 | 180 |
| Q4 | | 1392 | | 1881 | | 4271 | 1145 | 1096 |
| Q5 | | 6418 | | 9593 | | 29292 | 5548 | 6900 |
| Q6 | | 1965 | | 2537 | | 5436 | 1712 | 1740 |
| Q7 | | 728 | | 948 | | 1466 | 2548 | 706 |
| Q8 | | 1568 | | 2098 | | 4583 | 1417 | 1442 |
| Q9 | | 584 | | 706 | | 1737 | 533 | 521 |
| Q10 | | 7118 | | 8935 | | 22549 | 6280 | 6578 |
| Q11 | | 582 | | 624 | | 1233 | 567 | 546 |
| Q12 | | 3143 | | 4086 | | 10179 | 2694 | 2567 |
| Q13 | | 475 | | 545 | | 1156 | 525 | 483 |
| Q14 | | 3966 | | 5346 | | 11716 | 3479 | 3843 |
| Q15 | | 11768 | | 15145 | | 26160 | 9162 | 9831 |
| Q16 | | 13837 | | 14132 | | 29183 | 14091 | 14884 |
| Q17 | | 1489 | | 3792 | | 6837 | 2183 | 1312 |
| Q18 | | 30271 | | 38966 | | 96159 | 26971 | 27762 |
| Q19 | | 3586 | | 5041 | | 12990 | 3082 | 3313 |
| Q20 | | 906 | | 1153 | | 2347 | 776 | 744 |
| Q21 | | 1921 | | 2456 | | 5653 | 1698 | 1716 |
| Q22 | | 303 | | 361 | | 784 | 257 | 239 |
| Geometric mean (ms) | | 2284 | | 2992 | | 6747 | 2105 | 2092 |
| Duration per query set (s) | | 98 | | 126 | | 292 | 88 | 91 |
| Queries per hour (QphH) | | 804.2 × 1 | | 628.0 × 1 | | 270.8 × 4 | 895.9 × 1 | 863.4 × 8 |

**Table 2: First performance results obtained from PostgreSQL**

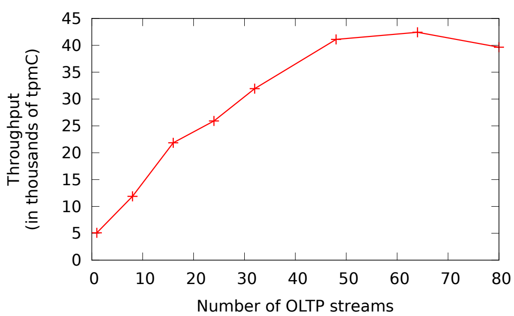



**Figure 3: Throughput with varying number of OLTP streams (no OLAP query streams)**

queries suffer more from the parallel execution than the OLTP transactions. These numbers also indicate that we cannot directly infer the results of the OLTP and OLAP parts run in isolation — even for a very simple workload where we add a single OLAP query stream to eight OLTP streams. Contention between workloads becomes worse as more streams are added to the workload.

# 6. FUTURE CHALLENGES

Although this paper describes a prototype of the CH-benCHmark, there are details that need to be worked out: First, we need to run more experiments with alternative isolation levels and data freshness to understand how the components of the mixed workload interact. Second, we must identify metrics that consider that the size of the data increases over time. Note that TPC-H does not consider a growing database over the course of a benchmark run. Third, we explore the possibility to add "ad-hoc" queries, i. e., to add queries with unpredictable data use.

# 7. CONCLUSIONS

In this paper, we presented the novel CH-benCHmark, a benchmark that executes a transactional and an OLAP query suite on a shared set of tables in parallel in the same database. The performance measurements when running the two parts in parallel cannot be easily inferred from the results of the TPC-C and TPC-H benchmarks. Consequently, we also identified metrics to quantify the performance of the mixed workload, to gauge how the different parts of the workload interact, and how to compare different systems.

# 8. REFERENCES

[1] A. Bog, K. Sachs, and A. Zeier. Benchmarking Database Design for Mixed OLTP and OLAP Workloads. In *ICPE'11*, 2011.

[2] A. Bog, J. Schaffner, and J. Krüger. A Composite Benchmark for Online Transaction Processing and Operational Reporting. In *IEEE Symposium on Advanced Management of Information for Globalized Enterprises (AMIGE)*, 2008.

[3] D. Feinberg and M. A. Beyer. Magic Quadrant for Data Warehouse Database Management Systems. `http://www.gartner.com/technology/media-products/reprints/teradata/vol3/article1/article1.html`, 2011.

[4] F. Funke, A. Kemper, and T. Neumann. Benchmarking Hybrid OLTP&OLAP Database Systems. In *Database Systems for Business, Technology and Web (BTW)*, 2011.

[5] M. Poess and C. Floyd. New TPC Benchmarks for Decision Support and Web Commerce. *SIGMOD Record*, 29(4):64–71, 2000.

[6] Transaction Processing Performance Council. TPC Benchmark C (Standard Specification), 2010.

[7] Transaction Processing Performance Council. TPC Benchmark H (Standard Specification), 2010.

[8] VoltDB. TPC-C-like Benchmark Comparison-Benchmark Description. `http://community.voltdb.com/node/134`.