Performance Tests

Performance Test Case TPC-B

TPC-B is a simple OLTP (online transaction processing) stress test which is useful for measuring the sheer performance of update operations.

The official TPC-B specification specifies strict test conditions and reporting requirements for the combination of software / hardware used for the tests. These tests should not be considered as TPC-B tests. The description and measurements follow:

The database consists of several bank branches, each with 10 tellers and 100,000 accounts. A transaction updates a row in each of the branch, teller and account tables, selects the updated row from the account table, and inserts a row into the history table. A total of 5 operations are performed per transaction.

The HyperSQL SVN repository contains the class org.hsqldb.test.TestBench used for these tests. We used a scale factor of 40, therefore there were 40 branches, 400 tellers and 4,000,000 accounts in the database. We performed all tests with 4 connections, each performing 8,000 transactions (32,000 transactions or 160,000 database operations per round). Each test was performed several rounds, one of which is reported here. A dual core 2.5 GHz computer (2008 spec) was used.

The maximum speed was 55,363 transactions per second (276,815 operations per second)

You are encouraged to run the tests for yourself and report the results to our Open Discussion Forum or Users mailing list. The test program works with other database engines and can be used for comparison.

The first test uses memory tables and no logging. This demonstrates the sheer speed of the multithreaded engine.

```
java -server -Xmx1536M org.hsqldb.test.TestBench -tps 40 -driver org.hsqldb.jdbcDriver -url jdbc:hsqldb:mem:test;hsqldb.tx=mvcc -user sa -init -
clients 4 -tpc 8000

* Benchmark Report *
--------------------
Time to execute 32000 transactions: 0.578 seconds.
Max/Min memory usage: 849712968 / 747964456
0 / 32000 failed to complete.
Transaction rate: 55363.32179930796 txn/sec.
```

The second test uses memory tables and logs the statement, performing automatic checkpoints (hsqldb.log_size=200) when the log reaches 200 MB. Logging the transactions takes about the same amount of time as performing them. Also when the checkpoint does occur, it slows down the particular test round.

```
java -server -Xmx1536M org.hsqldb.test.TestBench -tps 40 -driver org.hsqldb.jdbcDriver -url
jdbc:hsqldb:file:test;hsqldb.log_size=200;hsqldb.tx=mvcc -user sa -init -clients 4 -tpc 8000

* Benchmark Report *
--------------------
Time to execute 32000 transactions: 1.266 seconds.
Max/Min memory usage: 888403376 / 744425784
0 / 32000 failed to complete.
Transaction rate: 25276.46129541864 txn/sec.
```

The third test uses cached tables and logs the statements similar to the previous test.

```
java -server -Xmx128M org.hsqldb.test.TestBench -tps 40 -driver org.hsqldb.jdbcDriver -url
jdbc:hsqldb:file:test;hsqldb.default_table_type=cached;hsqldb.log_size=200;hsqldb.tx=mvcc -user sa -init -clients 4 -tpc 8000

* Benchmark Report *
--------------------
Time to execute 32000 transactions: 5.375 seconds.
Max/Min memory usage: 46327392 / 13014720
0 / 32000 failed to complete.
Transaction rate: 5953.488372093023 txn/sec.
```

January 2018 Update - tests on up-to-date hardware

131,147 transactions per second

The original tests were run in 2011 on a dual-core 2.5 GHz computer (2008 spec). For comparison, the first in-memory test was rerun in 2018 on a quad-core 4.4 GHz computer (2015 spec)

```
java -server -Xmx8000M org.hsqldb.test.TestBench -tps 40 -driver org.hsqldb.jdbcDriver -url jdbc:hsqldb:mem:test;hsqldb.tx=mvcc -user sa -init -
clients 4 -tpc 8000

* Benchmark Report *
--------------------
Time to execute 32000 transactions: 0.244 seconds.
Max/Min memory usage: 1099958 / 850497 kb
0 / 32000 failed to complete.
Transaction rate: 131147.54098360657 txn/sec.
1099958;850497;0;131147.54098360657
```

1,438,202 selects per second on up-to-date hardware

The latest version of TestBench supports the -select switch for testing only the SELECT query in TPC-B

```
java -server -Xmx8000M org.hsqldb.test.TestBench -tps 40 -driver org.hsqldb.jdbcDriver -url jdbc:hsqldb:mem:test;hsqldb.tx=mvcc -user sa -init -
clients 4 -tpc 64000

* Benchmark Report *
--------------------
Time to execute 256000 transactions: 0.178 seconds.
Max/Min memory usage: 891894 / 784390 kb
0 / 256000 failed to complete.
Transaction rate: 1438202.2471910112 txn/sec.
891894;784390;0;1438202.2471910112
```

Performance Test Case TPC-C

TPC-C is a more advanced OLTP test compared to TPC-B as there are different types of transactions for order placement, stock level query, etc. and each transaction queries, inserts, or updates multiple rows in several tables. However TPC-C was not designed as a stress tests because it emulates human-operated computer terminals with delays introduced between different stages of transactions. With the dominance of the World Wide Web, this access pattern no longer represents a realistic scenario. There is a Java test suite based on TPC-C here http://jtpcc.sourceforge.net/ and a modified version to use compiled statements here http://sourceforge.net/projects/benchmarksql/. We modified the original test and removed the delays, thus turning it into a stress test.

HyperSQL performed the TPC-C derivative test at 26,000 transactions per minute over a sustained period using memory tables without logging (20,000 with logging). The database was set up with 2 warehouses and eight clients were used in the test.

The test implementation does not look optimised and would certainly perform better (probably twice as fast) if it is modified to use compiled statements. We have not tried the second version.

It would be interesting to run the test documented here http://community.voltdb.com/node/134 which is used by VoltDB, a super-fast next generation RDBMS that actually incorporates the HyperSQL code for processing SQL statements.

See the Performance page for different configurations of HSQLDB and how they affect performance.