

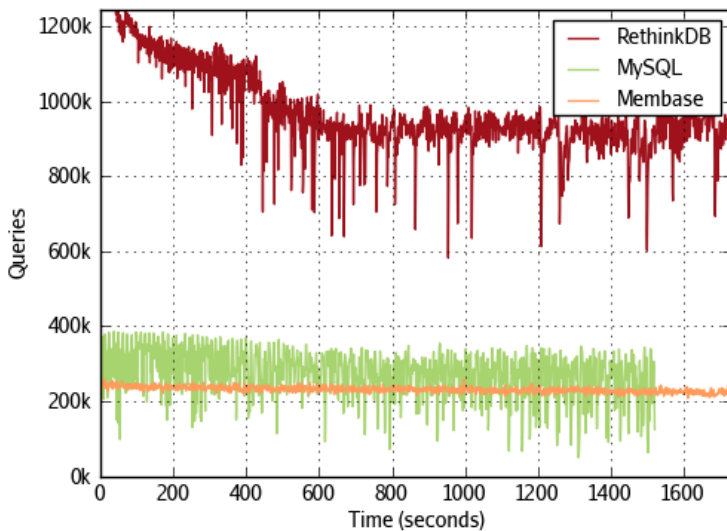


RethinkDB performance report.

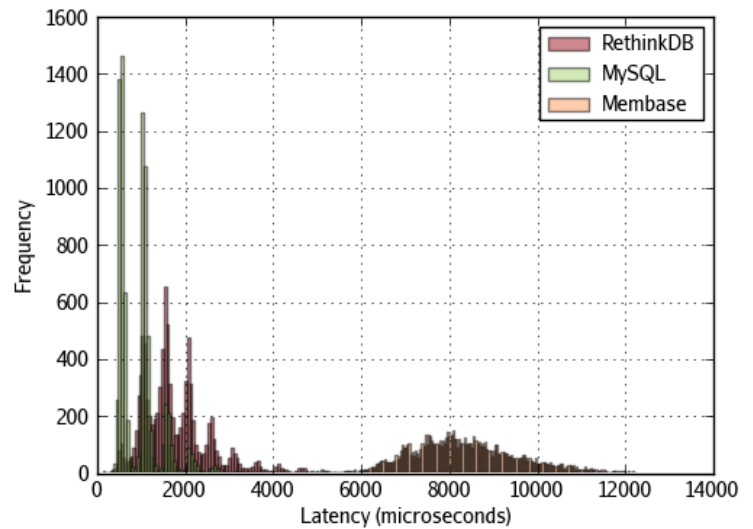
Canonical workload

The canonical workload simulates a common workload on typical hardware, as defined in the methodology section. The canonical workload reveals the performance of a database under a mixed workload, and provides a one-stop figure for overall database performance under typical conditions. Both MySQL and Membase provide stable performance at around 200k-300k queries per second. RethinkDB starts at more than 1.2M qps and settles into a stable throughput of about 900k qps after ten minutes.

Queries per second



Latency in microseconds

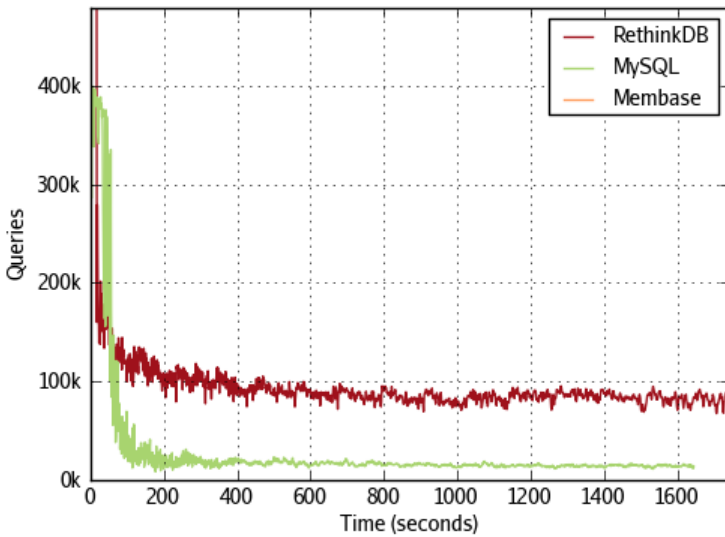


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	918,630.14	56,652.14	1,022,213.46	688,849.68	RethinkDB	2,099.94	3,497.26	6,751.62	499.72
Membase	228,320.05	6,666.16	243,286.88	213,086.49	Membase	8,490.27	1,286.18	12,115.34	6,073.20
MySQL	263,936.71	59,762.19	354,259.21	91,442.16	MySQL	1,792.39	34,625.63	4,167.70	435.52

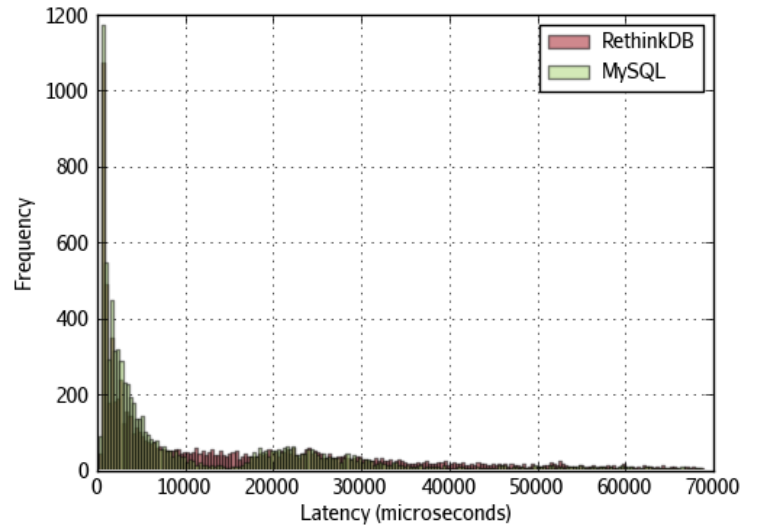
SSD I/O performance

This benchmark runs a canonical workload (as defined in the methodology section) under low memory conditions to simulate significant disk I/O. This benchmark provides insights into how different database systems behave when performance becomes almost entirely I/O bound. RethinkDB shows a ~5.5 factor improvement over MySQL. Controlling the cache size is not possible in Membase, so were unable to collect meaningful results for Membase's performance under these conditions.

Queries per second



Latency in microseconds

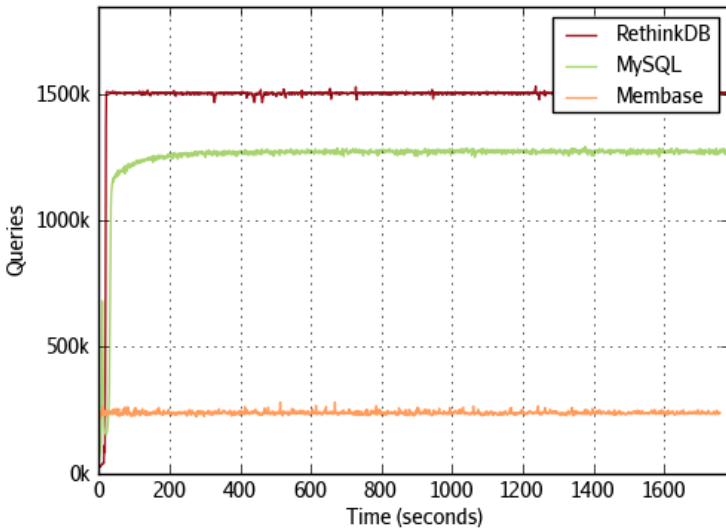


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	84,425.69	5,746.64	98,191.92	70,572.20	RethinkDB	23,721.63	30,014.29	130,574.56	494.05
Membase	N/A	N/A	N/A	N/A	Membase	N/A	N/A	N/A	N/A
MySQL	14,763.22	1,814.57	20,312.50	11,583.88	MySQL	26,220.17	65,108.00	414,120.75	499.63

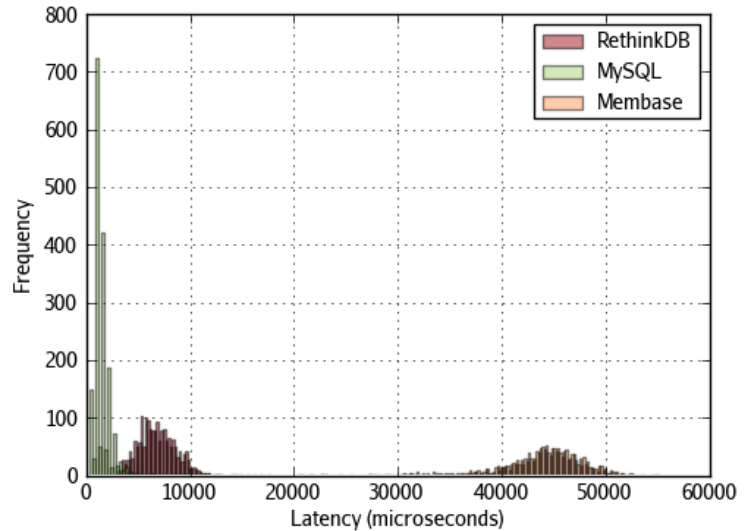
Pure select performance

In this benchmark, we test the database with a read-only workload. During the benchmark, data gets randomly selected from a database containing 50 million keys, each with a corresponding value size of 8-128 bytes. Due to the large cache size used in this benchmark (32GB), most of the data is loaded into memory within roughly the first 30 seconds. From that point on, the in-memory index structures and query processing capabilities determine the throughput actually available to the user. Both MySQL and RethinkDB provide high throughput rates of more than 1200k queries per second, with a steady throughput of 1500k qps for RethinkDB. Membase on the other hand is unable to achieve more than about 250k qps under this workload. Please note that (as in all benchmarks) we use prepared statements for MySQL to minimize the overhead of SQL query parsing. Both RethinkDB and Membase are accessed through the Memcached text protocol, which comes with comparably low parsing overhead out of the box.

Queries per second



Latency in microseconds



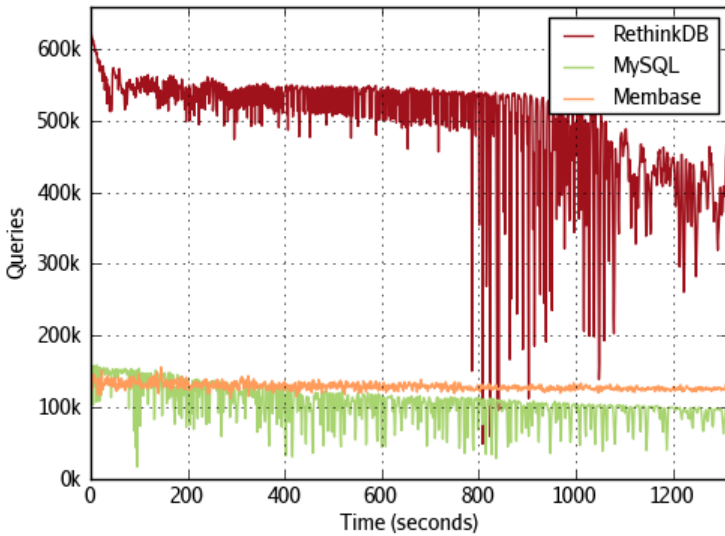
	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	1,504,630.15	6,758.96	1,523,583.14	1,474,791.00	RethinkDB	6,776.97	1,805.75	11,275.87	3,463.77
Membase	238,602.97	5,784.69	259,057.64	228,389.81	Membase	43,835.26	4,103.57	51,158.28	30,824.27
MySQL	1,273,951.04	5,477.09	1,285,149.74	1,259,006.35	MySQL	1,470.63	745.97	3,976.06	483.05

Pure insert performance

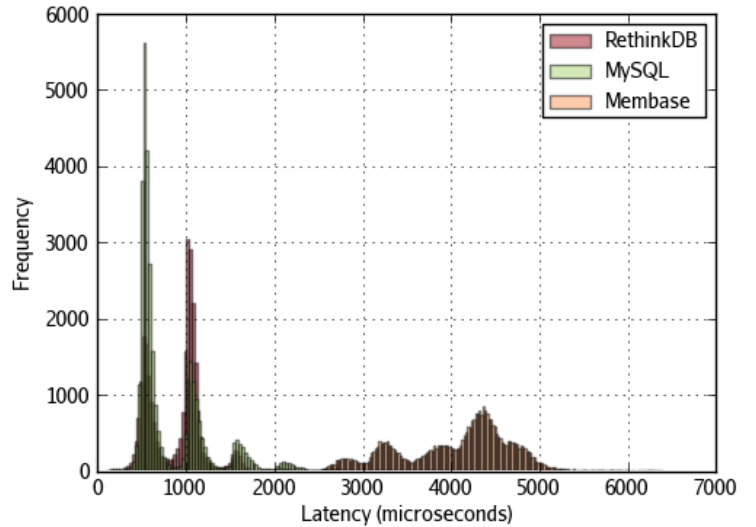
In this benchmark, we test the database with an insert-only workload. We continuously insert keys of sizes between 8 to 16 bytes with corresponding values of sizes between 8 and 128 bytes via a large number of concurrent connections. Both Membase and MySQL achieve a steady insert throughput of 100k - 150k queries per second. RethinkDB maintains an insertion rate of more than 500k keys per second for the first 750 seconds.

The eventual drop in RethinkDB's performance is due to the fact that the in-memory cache gets almost entirely filled with dirty pages, and disk throughput saturation is reached. The next version of RethinkDB will address this problem by making better use of locality (two random small modifications should result in a single small write, not two full block writes).

Queries per second



Latency in microseconds

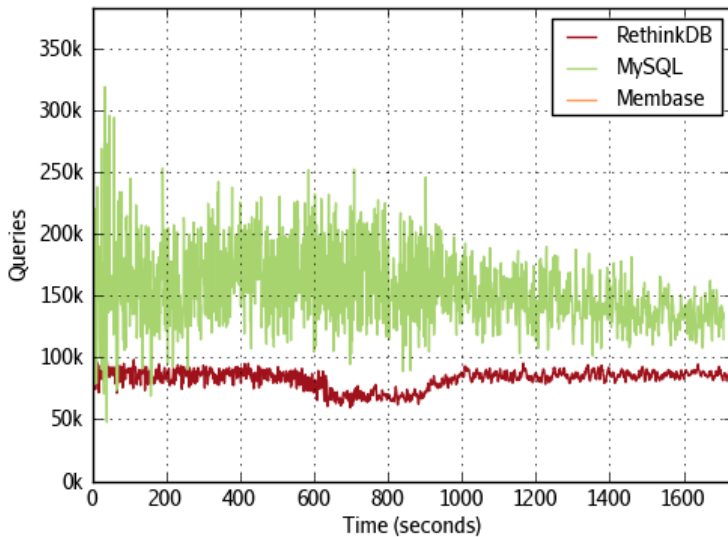


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile	Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	478,044.29	81,908.26	549,289.82	180,732.32	1,061.60	12,044.57	2,701.81	391.32
Membase	125,161.64	3,701.89	134,873.98	116,903.78	4,093.05	609.54	5,264.95	2,716.10
MySQL	93,762.94	17,181.36	122,000.94	38,777.84	1,062.17	13,116.03	2,728.48	420.63

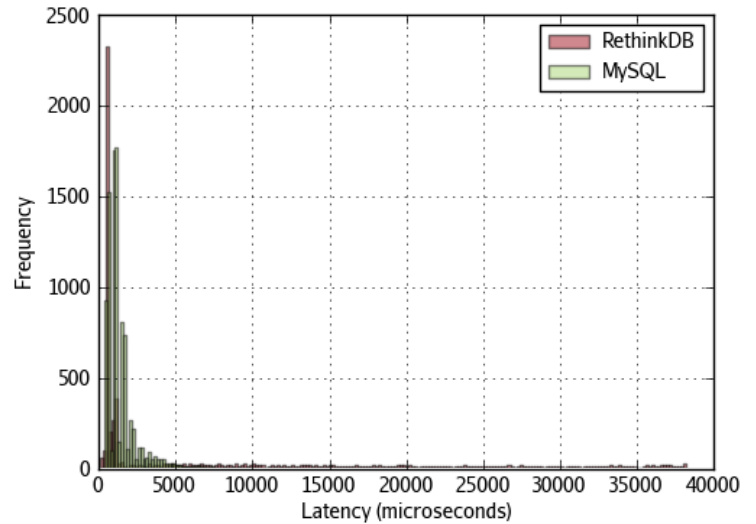
Strong durability guarantees

Running in strong durability mode guarantees that whenever a write operation returns a success status to the client, the change has actually been committed to non-volatile memory. Due to limitations in RethinkDB's current implementation and its transaction log-free architecture, RethinkDB is unable to provide performance figures on-par with MySQL for this workload. The next version will address this issue and provide significantly higher performance under strong durability guarantees.

Queries per second



Latency in microseconds

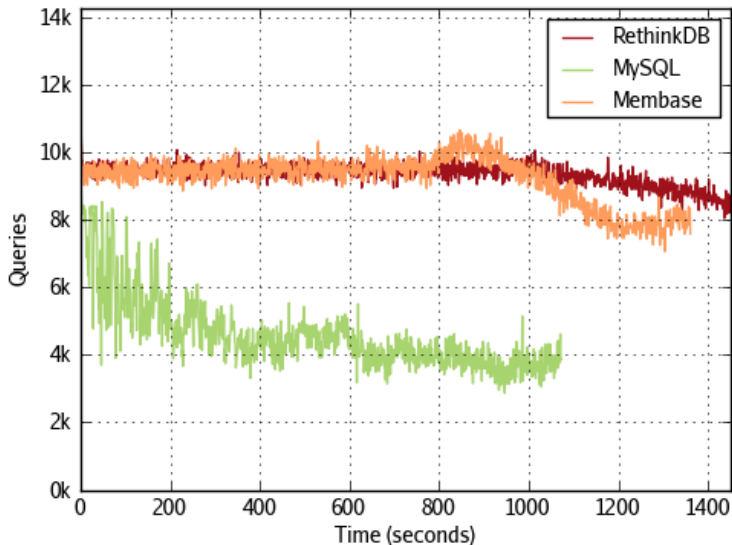


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	81,295.60	7,670.12	92,319.96	62,885.18	RethinkDB	23,933.57	32,912.33	71,075.92	451.19
Membase	N/A	N/A	N/A	N/A	Membase	N/A	N/A	N/A	N/A
MySQL	152,028.09	25,871.99	224,750.59	107,138.35	MySQL	2,681.66	17,843.98	18,748.99	464.17

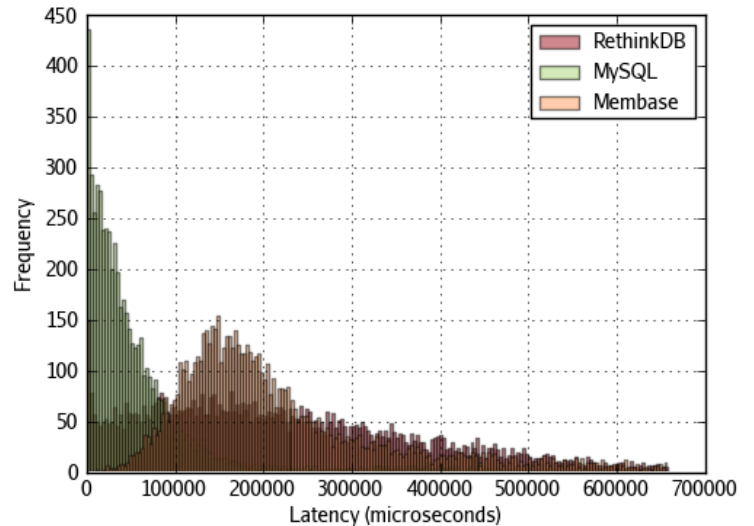
Canonical workload with large values (4KB–64KB)

The large values benchmark tests database performance for workloads involving values of sizes between 4KB and 64KB (evenly distributed across sizes in this range). Both Membase and RethinkDB provide a mostly stable throughput of around 10k queries per second. The throughput is limited due to network bandwidth saturation in our setup. However, this benchmark reveals a performance problem of MySQL for this workload, which only provides a steady throughput of around 4k qps.

Queries per second



Latency in microseconds

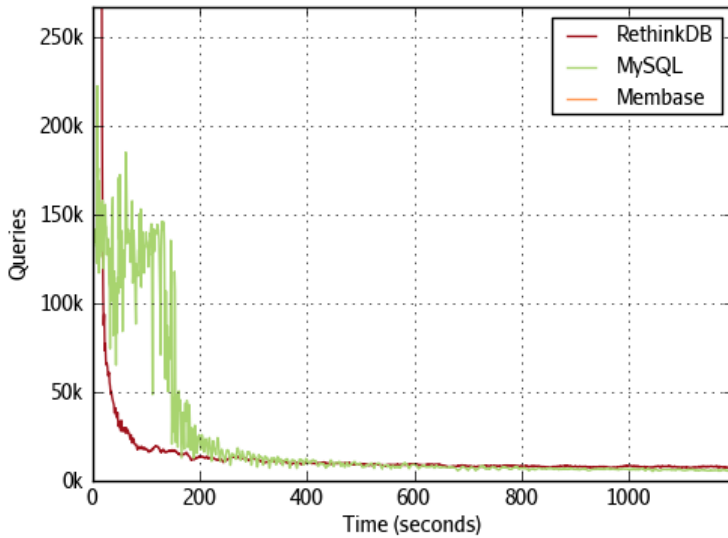


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	9,321.40	328.43	9,872.64	8,377.80	RethinkDB	250,878.42	200,992.23	922,713.05	2,118.26
Membase	9,167.12	784.25	10,445.00	7,488.60	Membase	260,979.93	208,269.93	1,122,080.16	49,926.93
MySQL	4,145.75	498.35	5,386.16	3,152.92	MySQL	155,595.86	462,500.52	2,692,923.24	1,201.51

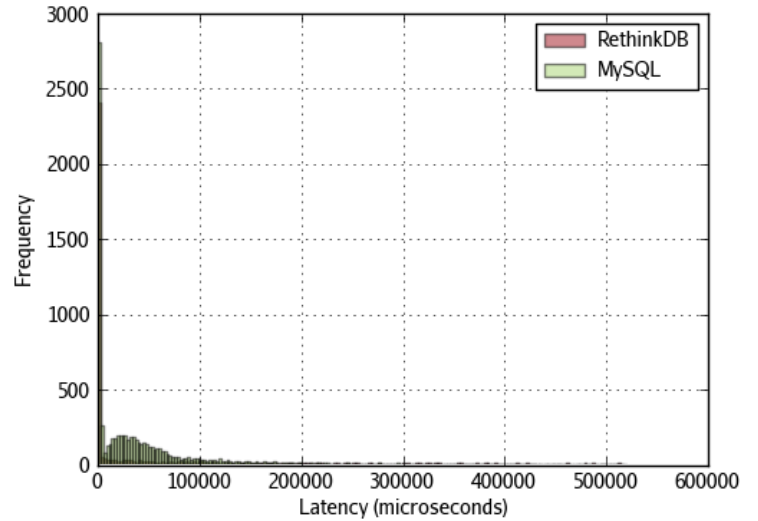
Rotational drive I/O performance

This benchmark runs a canonical workload under high cache pressure with the backing store located on rotational hard drives. RethinkDB demonstrates a slightly higher throughput than MySQL, at the expense of higher latency.

Queries per second



Latency in microseconds

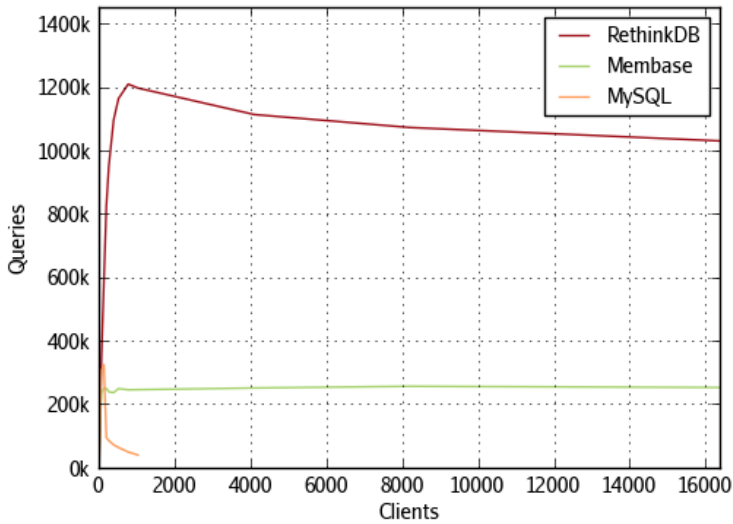


	Mean qps	Standard deviation	Upper 1-percentile	Lower 1-percentile		Mean latency	Standard deviation	Upper 1-percentile	Lower 1-percentile
RethinkDB	8,279.01	1,030.13	11,662.22	6,952.96	RethinkDB	381,892.86	617,492.62	1,941,667.72	230.44
Membase	N/A	N/A	N/A	N/A	Membase	N/A	N/A	N/A	N/A
MySQL	6,415.41	1,389.84	10,935.74	4,664.38	MySQL	68,048.02	128,704.24	425,758.35	484.69

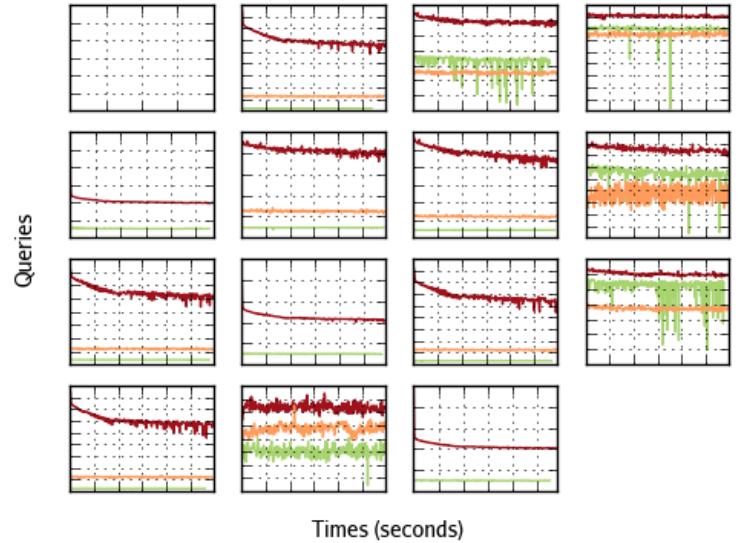
Concurrent connections

In this benchmark, we test the database's ability to scale to a large number of concurrent connections. We run a canonical workload on a various number of concurrent connections (from one connection, to many thousands), and measure corresponding performance. Here it is evident that RethinkDB and Membase scale across many connections linearly, while MySQL is not capable of efficiently supporting more than 256 connections.

Average queries per second across runs



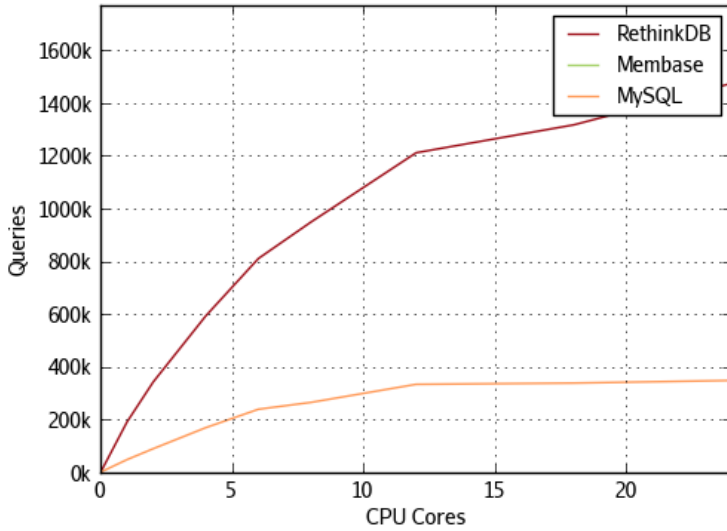
Queries per second across runs



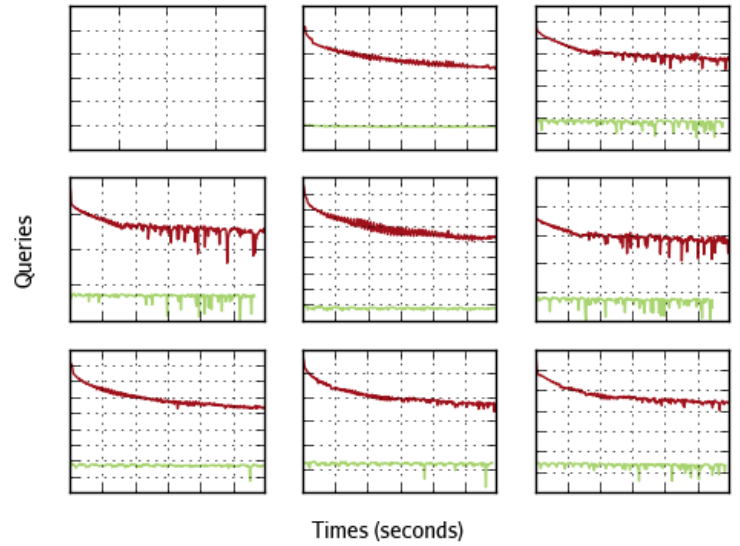
Multicore scalability

Here we test the database's ability to scale across multiple cores. We run the canonical workload on databases configured to use anywhere from one to thirty two CPU cores.

Average queries per second across runs



Queries per second across runs



Appendix to performance report.

Setup and methodology

There are dozens of potential software tuning parameters, user workloads, and hardware configurations. If we attempt to benchmark most of these combinations, the number of possible benchmarks exhibits exponential growth and quickly becomes infeasible. Therefore, we must restrict the number of benchmarks to a limited set that is interesting to a variety of users. We accomplish this by defining a *canonical configuration* and a *canonical workload* as follows:

Canonical workload:

- All benchmarks are limited to performing storage, retrieval, modification, and deletion of key/value pairs.
- Membase and RethinkDB are accessed via the Memcached protocol. MySQL is accessed via the prepared statements C API.
- Read/write ratios - 1 delete per 4 updates, per 8 inserts, per 64 reads.
- 512 client connections.
- Key sizes randomly distributed between 8 and 16 bytes.
- Value sizes randomly distributed between 8 and 128 bytes.
- Selects are randomly batched between 1 and 16 operations per query.

Canonical configuration:

- Two hexacore Intel CPUs.
- 32 GB of RAM.
- Four X25-M Intel SSDs.
- Two 1GbE network cards.
- The load-generating stress client is run on a separate machine with two 1GbE network cards. We make sure that the stress client does not present CPU or disk bottlenecks.
- We use MySQL 5.1.50-1 with Facebook patches (with InnoDB engine), and Membase server 1.6.0.1 (community edition) for all testing.
- Competitors are set up to mirror functionality as closely as possible. In order to simulate key/value storage, MySQL is configured with a two column InnoDB table. Each column is a varchar (limited to maximum key size and maximum value size), and the keys are indexed via a primary key.

We set up benchmarks to measure canonical workloads on canonical hardware. In order to expose special cases, we adjust these two configurations to a given benchmark as necessary. Specific parameters for each benchmark are described below.

Canonical workload

RethinkDB:

```
Server parameters:  -c 12 -m 32768 -f /dev/sdf -f /dev/sdg -f /dev/sdh -f /dev/sdi
```

Membase:

```
Server parameters:  -d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data --innodb_log_file_size=1G
```

Pure select performance

RethinkDB:

```
Server parameters:  -c 12 -m 32768
```

Membase:

```
Server parameters:  -d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```

Pure insert performance**RethinkDB:**

```
Server parameters:  -c 12 -m 32768
```

Membase:

```
Server parameters:  -d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```

Strong durability guarantees**RethinkDB:**

```
Server parameters:  --wait-for-flush y --flush-timer 50 -c 12 -m 32768 -f /dev/sdf -f /dev/sdg -f /dev/sdh -f
/dev/sdi
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=1 --innodb_doublewrite=1 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```

Canonical workload with large values (4KB-64KB)**RethinkDB:**

```
Server parameters:  --active-data-extents 1 -c 12 -m 32768
```

Membase:

```
Server parameters:  d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
```

```
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```

Rotational drive I/O performance

RethinkDB:

```
Server parameters:  --active-data-extents 1 -c 12 -m 256 -f /dev/sdb -f /dev/sdc -f /dev/sdd -f /dev/sde
```

Membase:

```
Server parameters:  -d membase_data -m 256
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=/tmp/raidtest --innodb_data_home_dir= --
innodb_data_file_path=/dev/sdb:20Graw --innodb_buffer_pool_size=256m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0
```

Concurrent connections

RethinkDB:

```
Server parameters:  -c 12 -m 32768
```

Membase:

```
Server parameters:  -d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```

Multicore scalability

RethinkDB:

```
Server parameters:  -c 12 -m 32768
```

Membase:

```
Server parameters:  -d membase_data -m 32768
```

MySQL:

```
Server parameters:  --disable-log-bin --skip-name-resolve --query_cache_size=0 --transaction_isolation=READ-
UNCOMMITTED --innodb_support_xa=off --innodb_lock_wait_timeout=5 --
innodb_flush_method=O_DIRECT --max-connections=1030 --max-prepared-stmt-count=32764 --
innodb_log_group_home_dir=mysql_data --innodb_buffer_pool_size=32768m --
innodb_flush_log_at_trx_commit=0 --innodb_doublewrite=0 --innodb_data_home_dir=mysql_data
--innodb_log_file_size=1G
```