BENCHMARK: PRELIMINARY RESULTS

JUNE 25, 2014

Our latest benchmark test results are in. The detailed report will be published early next month, but after 6 weeks of designing and running these tests we're excited to share the preliminary results.

THE TITLE

Scaling popular NoSQL databases with memory-bound workloads: A Thumbtack study

THE TEST

We took three of the most popular NoSQL databases (DataStax Enterprise 2.0.7, Couchbase 2.5.1, and MongoDB 2.6.1) and measured how much load we could put through them while keeping the working set in RAM. The reason for this test design was to measure the upper bounds on how an application might perform when the load is demanding but there is tolerance to losing a small window of recent updates under total node failures. These kinds of requirements are very common in things like pre-warmed caches, distributed web session stores, user profile matching, and real-time bidding.

Our hypothesis was that Couchbase would handle an order of magnitude more throughput in this use case. Couchbase has its roots in memcached, which is a proven cache layer whose strong in-memory performance characteristics are well known. If Couchbase was able to meet its promise of preserving close to memcached performance while providing persistence, replication, and failover, it would be difficult for the other systems to surpass in raw throughput as long as most of the working set is mostly in RAM. We also expected both MongoDB and Cassandra to be able to serve tens of thousands of requests per second.

THE CONFIGURATION

A challenge in doing these kinds of studies is that Cassandra, Couchbase, and MongoDB have different storage and replication architectures. However, they each have a built-in cache layer which allows them to be used as scalable distributed cache storage with eventual persistence and replication.

Our test attempted to optimize each system's performance for the case where all data could fit into memory, and to use the fastest consistency and durability modes available for the particular database. To simulate load, we used a version of YCSB that incorporated our own changes to distribute load as well as optimizations from various vendors.

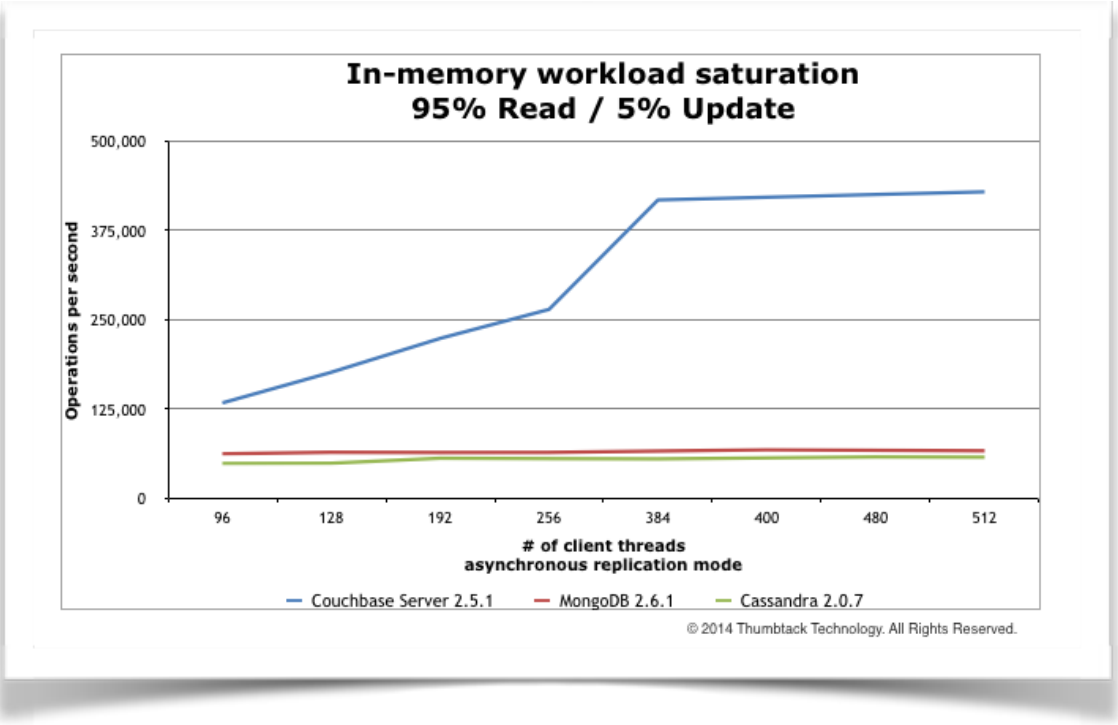
We used 4 Xeon E5-2620 barebone server nodes to host database clusters and numerous cloud-based virtual client nodes for generating the workload. The number of client nodes and threads was independently tuned to yield the highest performance on each database. Each server node was fine-tuned on the HW/OS-level according to recommendations from each of the vendors in the study. We gathered numerous low-level metrics from client and server machines using Graphite throughout testing process, but focus on presenting the operational statistics.

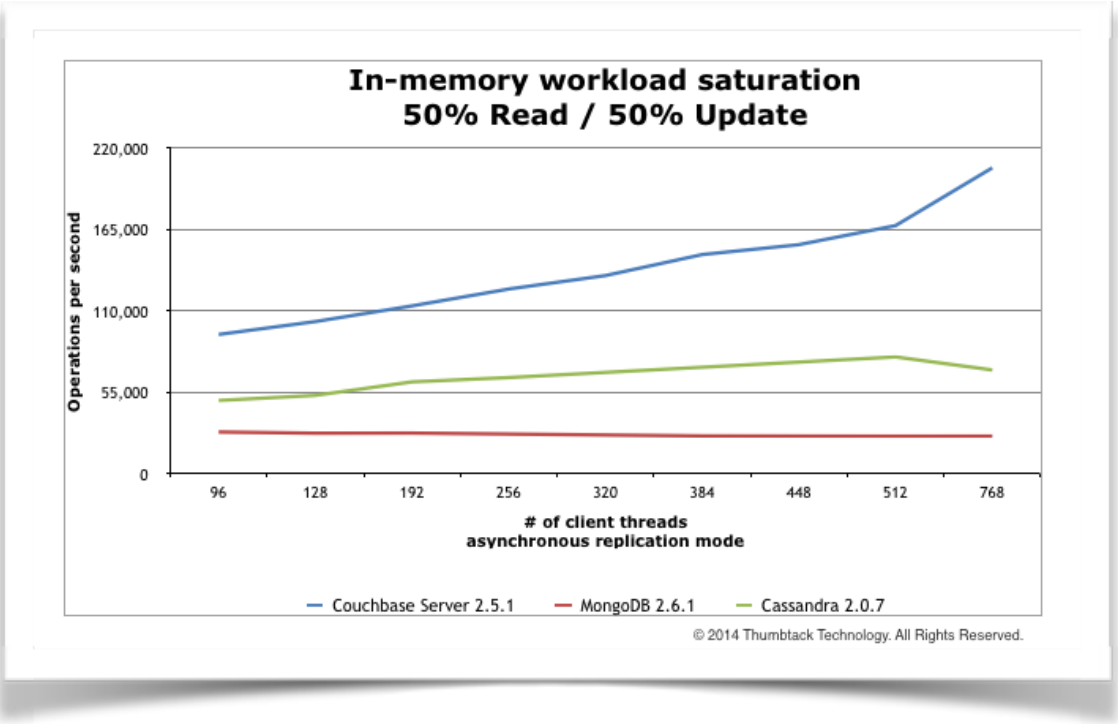
THE TEST DESIGN

For each database a 20M record dataset was generated, using half the available RAM. We experimented with various numbers of client nodes (up to 16) and client threads (up to 32 per node) to get the highest throughput with reasonable latency. We tested two workloads emulating a Heavy-Read use case with 95% reads and 5% updates, and a Heavy-Write case with 50% reads and 50% updates. We chose a Zipfian distribution to simulate "hot spots" of active users.

THE PRELIMINARY RESULTS

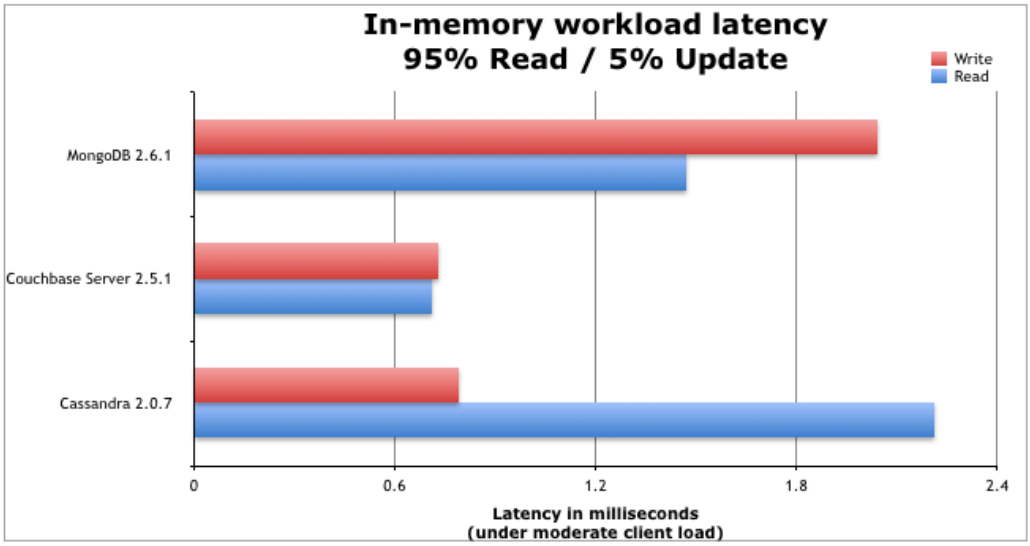
These results represent the case where 90% of the dataset and its replica fit into memory, with very small I/O usage of the disk subsystem. All databases were configured for asynchronous persistence and replication.

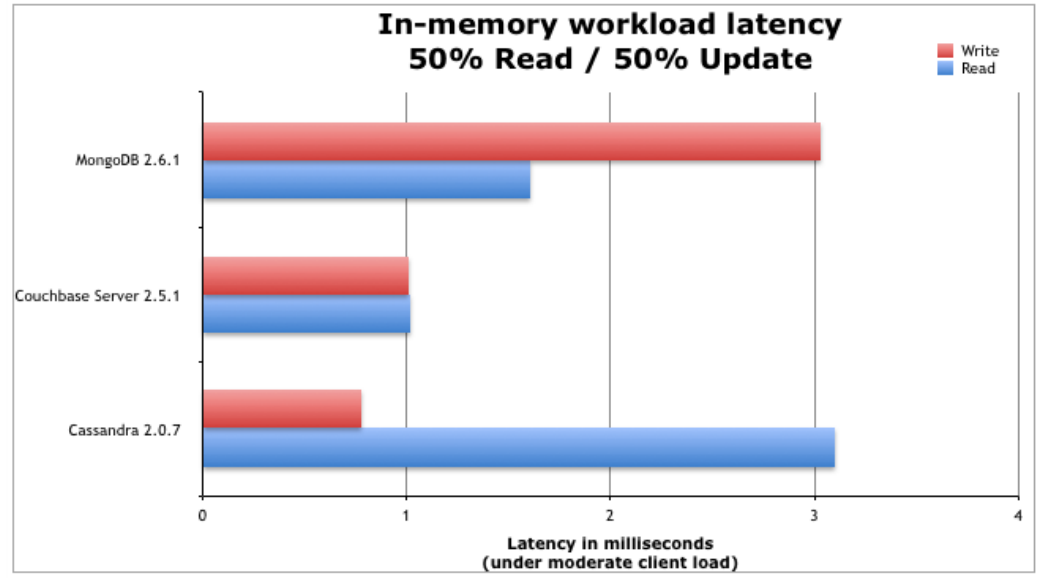




THE PRELIMINARY RESULTS

continued





THE DETAILED REPORT

In the full report to be released next month we will provide a detailed description of the testing environment, intermediate results and in-depth observations. Also, we will report benchmark results for bigger database clusters and dataset sizes to determine how linear these databases can be scaled within the bounds of in-memory cache use cases.

The second part of the study examines how linearly these systems scale as new nodes are added to the cluster. Our hypothesis is that all three systems should scale close to linearly, at least across small cluster sizes. These results will be released with the rest of the report.

PREVIOUS NOSQL WHITE PAPERS BY THUMBTACK TECHNOLOGY

Ultra-High Performance NoSQL Benchmarking: Analyzing Durability & Performance Tradeoffs

NoSQL Failover Characteristics: Aerospike, Cassandra, Couchbase, and MongoDB

ABOUT THUMBTACK

Thumbtack Technology has been at the forefront of large-scale, high-performance web development for over a decade, setting the bar for both backend excellence and innovative applications. Our American-Siberian team of over a hundred business strategists and math-oriented engineers work in orchestrated harmony to bring high-speed, high-volume capacity to meet the most demanding business goals.

Collaborating on hundreds of high-profile web and mobile products has driven us to cultivate expertise in NoSQL deployment and migration, email and subscription platforms, ad serving and RTB, cloud computing and hosting. The cornerstone of Thumbtack Technology's success is the possession of deep knowledge across platforms, frameworks, and languages, enabling us to identify the precise set of tools which will make each endeavor a quantifiable success.

Client roster includes The New York Times, eBay, Shutterstock, AppNexus, The Boston Globe, Ad Marketplace, Etsy, OkCupid, SAP, Gazprom, Grubhub and PlaceIQ.