AMAZON ELASTICACHE

In-Memory Processing in the Cloud

Amazon ElastiCache Enables High Performance, Low Latency to Power Real-Time Applications with Sub-Millisecond Latency

Developers need a way to maintain super low latency even as they accommodate spikes in demand, and while controlling infrastructure and database costs.

Applications have become an indispensable part of our everyday lives. We have integrated them into our work, hobbies, shopping, and studying. Just as we live in real time, we expect our applications to operate in real time. When we tap or click on an application icon, we expect it to open immediately and for that immediacy to continue throughout the session, regardless of the complexity of the action. We also expect that while using an application our information is private and secure, and that this security does not interfere with the application's performance.

Users demand real-time applications. Read-heavy application workloads (such as social networking, gaming, media sharing, and Q&A portals) and compute-intensive workloads (such as a recommendation engine) must be built for ultra-low latency and high throughput.

Building applications for latency-free performance during a steady state is a challenge in and of itself. But developers must also anticipate sudden spikes in demand that increase the load on the database—and significantly degrade application performance. Handling these throughput spikes is a significant challenge for developers today. Organizations can't afford to miss out on the additional traffic—nor can they afford the impact of a negative user experience.

Sometimes, in an effort to support rapid growth and high-performance requirements, IT organizations over-provision database resources. This is both inefficient and costly. As the application scales, infrastructure and database costs increase, and the return on investment decreases. Developers need a way to maintain super low latency even as they accommodate spikes in demand, and while controlling infrastructure and database costs. The rest of this white paper explains how organizations can do just that.

The Benefits of In-Memory Data Stores

The strict performance requirements imposed by real-time applications mandate more efficient databases. Traditional databases rely on disk-based storage. A single user action may consist of multiple database calls. As they accumulate, latency increases. However, by accessing data in memory, in-memory data stores provide higher throughput and lower latency. In fact, in-memory data stores can be one to two orders of magnitude faster than disk-based databases.

Using managed cloud services also eliminates the need to administer infrastructure. Database hotspots are reduced and performance becomes more predictable. Some cloud-based services also offer the benefit of high availability, with replicas and support for multiple availability zones.

As a NoSQL data store, in-memory data stores do not share the architectural limitations found in traditional relational databases. NoSQL data stores are built from the ground up to be scalable. Traditional relational databases use a rigid table-based architecture. Some NoSQL data stores use a key-value store, and therefore don't enforce a structure on the data. This enables scalability and makes it easier to grow, partition, or shard data as data stores grow.

When consumed as a cloud-based service, an in-memory data store also provides availability and cost benefits. On-demand access allows organizations to scale their applications as needed in response to demand spikes, and at a lower cost than disk-based stores. Using managed cloud services also eliminates the need to administer infrastructure. Database hotspots are reduced and performance becomes more predictable. Some cloud-based services also offer the benefit of high availability, with replicas and support for multiple availability zones.

The Benefits of Distributed Cache

A caching layer helps further drive throughput for read-heavy applications. A caching layer is a high-speed storage layer that stores a subset of data. When a read request is sent, the caching layer checks to see if it has the answer. If it doesn't, the request is sent on to the database. Meeting read requests through the caching layer in this manner is more efficient and delivers higher performance than what can be had from a traditional database alone. It is also more cost effective.

A single node of in-memory cache can deliver the same read throughput as several database nodes. Instead of provisioning additional instances of your traditional database to accommodate a demand spike, you can drive more throughput by adding one node of distributed cache, replacing several database nodes. The caching layer saves you money because you're paying for one node instead of multiple database nodes, and you get the added benefit of dramatically faster performance for reads.

Amazon ElastiCache for Redis on Intel® Nodes

Amazon ElastiCache is a web service that makes it easy to deploy, operate, and scale an in-memory data store or cache in the cloud. The service improves application performance by allowing developers to retrieve information from fast, managed, in-memory data stores, instead of relying on slower disk-based databases. Amazon ElastiCache supports two open-source in-memory engines: Redis and Memcached. The primary use case for Memcached is caching; it is easy to use and scale. ElastiCache is protocol-compliant with Memcached, so tools used with existing Memcached environments work seamlessly with ElastiCache.

Redis is a leading in-memory NoSQL data store that supports persistence, availability, and Lua scripting. It comes with a set of versatile in-memory data structures that make it easy to create a variety of custom applications. Redis is often used for caching, session management, pub/sub, and leaderboards. Due to its speed and ease of use, Redis is a popular choice for web, mobile, gaming, ad-tech and IoT applications that require best-in-class performance. It is written in optimized C code and supports multiple development languages.

As a fully managed and optimized in-memory data store service, Amazon ElastiCache for Redis delivers the ease of use and power of Redis along with

ElastiCache for Redis can be used as a primary in-memory key-value data store, providing fast, sub-millisecond data performance, high availability, and scalability.

the availability, reliability, and performance required for real-time applications. Amazon ElastiCache for Redis utilizes an end-to-end optimized stack running on customer-dedicated nodes to provide both performance and security. ElastiCache for Redis can be used as a primary in-memory key-value data store, providing fast, sub-millisecond data performance, high availability, and scalability.

ElastiCache for Redis offers a variety of node types varying in in-memory capacity, network support, and performance, all built on Intel® Xeon® processors. ElastiCache nodes running on the latest Intel® Xeon® processors drive higher throughput. According to a benchmark, ElastiCache for Redis delivers 34% greater throughput across various Redis commands using M4 instances versus M3 instances. M3 is based on Intel® Xeon® E5-2670 v2 (Ivy Bridge) processors*; M4 is based on 2.3 GHz Intel® Xeon® E5-2686 v4 (Broadwell) processors or 2.4 GHz Intel® Xeon® E5-2676 v3 (Haswell) processors. M5 is based on 2.5 GHz Intel® Xeon® Platinum 8175 processors with new Intel® Advanced Vector Extension (Intel® AVX-512) instruction set.

Features and Benefits of Amazon ElastiCache for Redis Running on Intel® Processors

As a fully managed service, Amazon ElastiCache for Redis eliminates the overhead associated with running an in-memory data store in-house. IT no longer has to perform management tasks like hardware provisioning, software patching, setup, configuration, monitoring, failure recovery, and backups. Instead, staff can focus on those parts of the application that differentiate the business. This enables the better use of scarce IT resources, both in terms of infrastructure and staff. Organizations not only improve load and response times but also reduce the cost associated with scaling applications.

Amazon ElastiCache for Redis is also easily scalable and highly available. Organizations can start small and scale Redis data as their applications grow—all the way up to terabytes of in-memory data. Read capacity can be further scaled by adding read-replicas. High availability is also provided through multiple Availability Zones with automatic failover. ElastiCache automatically detects and replaces failed nodes, and provides a resilient system that mitigates the risk of overloaded databases, which slow website and application load times.

Integration with Amazon CloudWatch monitoring provides visibility into key performance metrics associated with your nodes, enabling organizations to quickly diagnose and react to issues. For example, organizations can set up thresholds and receive alarms if one of the nodes becomes overloaded with requests.

Amazon ElastiCache for Redis also delivers the high performance and throughput benefits organizations seek from an in-memory data store. One node of ElastiCache can offer hundreds of thousands—sometimes up to a million—calls per second—that's one to two orders of magnitude more than a disk-based database. The latency for a call to ElastiCache can be 300-500 microseconds compared to double-digit milliseconds for a traditional database.

A microsecond is one millionth of a second. A millisecond is one thousandth of a second.

"It's easy to setup, offers blazing fast performance, and gives users all the benefits that Redis has to offer."

Joel Hensley
Engineering Director, Hudl

ElastiCache in Action

Hudl is a software company that offers a sports platform for coaches, athletes, and analysts to improve gameplay through video analytics. The company's namesake application provides users with tools to edit and share video, study associated play diagrams, and create quality highlight reels for entertainment and recruiting.

When coaches and athletes log into Hudl, the first screen they see is their personalized news feed. The news feed is the first impression for users, and performance is critical. In addition, Hudl's traffic is highly seasonal, with football season being prime time for the application. To help deliver different data types while maintaining high performance, Hudl decided to move to Redis.

Hudl decided to give Redis a try with Amazon ElastiCache for Redis. Within minutes, Hudl had its first test node spun up running Redis. The organization was able to easily configure a Redis deployment, node size, and security groups, and use Amazon CloudWatch to monitor key metrics. What's more, the organization was able to create separate test and production clusters without waiting for an infrastructure engineer to manually set up and configure the servers.

Hudl launched the news feed on Amazon ElastiCache for Redis in April 2015. Starting around August, coaches and athletes from all over the country got back into football mode and logged into Hudl daily. During the week of Sept. 5-11, there were 1.2 million unique users accessing their feeds. The feed service averaged 300 requests per second, with a peak of 800. Here are some quick stats from ElastiCache during that week:

- **Total cached items:** 21 million
- **Cache hits:** 175k/min on average; 350k/min during peak times
- **Network in:** 43 MB/min on average; 101 MB/min during peak times
- **Network out:** 600 MB/min on average; 1.25 GB/min during peak times

Based on this success, ElastiCache for Redis quickly became their default option for caching. In the past year, five other key services at Hudl were moved to Redis on ElastiCache.

"It's easy to setup, offers blazing fast performance, and gives users all the benefits that Redis has to offer," says Joel Hensley, engineering director at Hudl.

Conclusion

In-memory data stores deliver the high performance and throughput that real-time applications require. Amazon ElastiCache for Redis is a web service optimized on Intel® Xeon® processors that makes it easy to deploy, operate, and scale an in-memory data store or cache in the cloud. Organizations get up and running quickly, and reap all the performance benefits of in-memory data stores while eliminating the overhead associated with database management.

To try Amazon ElastiCache for free, visit https://aws.amazon.com/elasticache

* Throughput determined by Requests per Second. Commands: Average % across SET, GET, INCR, LPUSH, RPUSH, LPOP, RPOP, SADD, SPOP, LPUSH, LRANGE_100, LRANGE_300, LRANGE_500, MSET was used.

Load: 1 Million Requests with Pipelined commands, Payload 3 bytes; Redis Instance compared: cache.m4.xlarge, cache.m3.xlarge; Same instance used to issue requests (c4.xlarge with Redis Client/benchmark tool installed).