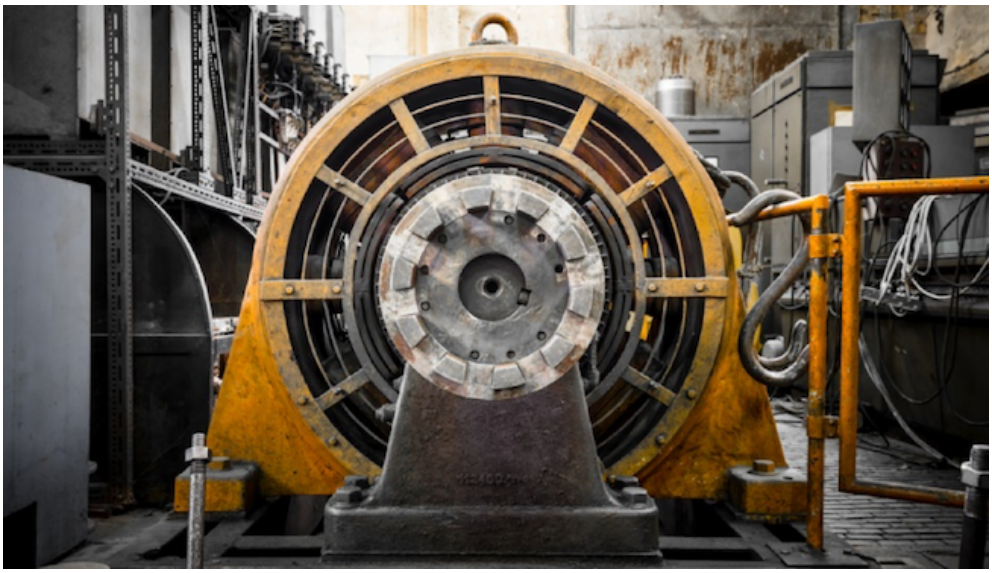
All Things Distributed

Werner Vogels' weblog on building scalable and robust distributed systems.

A Decade of Dynamo: Powering the next wave of high-performance, internet-scale applications

By Werner Vogels on 02 October 2017 11:00 AM | Permalink | Comments (16)



Today marks the 10 year anniversary of Amazon's Dynamo whitepaper, a milestone that made me reflect on how much innovation has occurred in the area of databases over the last decade and a good reminder on why taking a customer obsessed approach to solving hard problems can have lasting impact beyond your original expectations.

It all started in 2004 when Amazon was running Oracle's enterprise edition with clustering and replication. We had an advanced team of database administrators and access to top experts within Oracle. We were pushing the limits of what was a leading commercial database at the time and were unable to sustain the availability, scalability and performance needs that our growing Amazon business demanded.

Our straining database infrastructure on Oracle led us to evaluate if we could develop a purpose-built database that would support our business needs for the long term. We prioritized focusing on requirements that would support high-scale, mission-critical services like Amazon's shopping cart, and questioned assumptions traditionally held by relational databases such as the requirement for strong consistency. Our goal was to build a database that would have the unbounded scalability, consistent performance and the high availability to support the needs of our rapidly growing business.

A deep dive on how we were using our existing databases revealed that they were frequently not used for their relational capabilities. About 70 percent of operations were of the key-value kind, where only a primary key was used and a single row would be returned. About 20 percent would return a set of rows, but still operate on only a single table.

With these requirements in mind, and a willingness to question the status quo, a small group of distributed systems experts came together and designed a horizontally scalable distributed database that would scale out for both reads and writes to meet the long-term needs of our business. This was the genesis of the Amazon Dynamo database.

The success of our early results with the Dynamo database encouraged us to write Amazon's Dynamo whitepaper and share it at the 2007 ACM Symposium on Operating Systems Principles (SOSP conference), so that others in the industry could benefit. The Dynamo paper was well-received and served as a catalyst to create the category of distributed database technologies commonly known today as "NoSQL."

Contact Info

Werner Vogels
CTO - Amazon.com

werner@allthingsdistributed.com

Other places

Follow werner on twitter if you want to know what he is current reading or thinking about.
At werner.ly he posts material that doesn't belong on this blog or on twitter.

Check out some great music

Syndication

Subscribe to this weblog's atom feed or rss feed

Archives

All postings

Of course, no technology change happens in isolation, and at the same time NoSQL was evolving, so was cloud computing. As we began growing the AWS business, we realized that external customers might find our Dynamo database just as useful as we found it within Amazon.com. So, we set out to build a fully hosted AWS database service based upon the original Dynamo design.

The requirements for a fully hosted cloud database service needed to be at an even higher bar than what we had set for our Amazon internal system. The cloud-hosted version would need to be:

- **Scalable** – The service would need to support hundreds of thousands, or even millions of AWS customers, each supporting their own internet-scale applications.
- **Secure** – The service would have to store critical data for external AWS customers which would require an even higher bar for access control and security.
- **Durable and Highly-Available** – The service would have to be extremely resilient to failure so that all AWS customers could trust it for their mission-critical workloads as well.
- **Performant** – The service would need to be able to maintain *consistent* performance in the face of diverse customer workloads.
- **Manageable** – The service would need to be easy to manage and operate. This was perhaps the most important requirement if we wanted a broad set of users to adopt the service.

With these goals in mind, In January, 2012 we launched Amazon DynamoDB, our cloud-based NoSQL database service designed from the ground up to support extreme scale, with the security, availability, performance and manageability needed to run mission-critical workloads.

Today, DynamoDB powers the next wave of high-performance, internet-scale applications that would overburden traditional relational databases. Many of the world's largest internet-scale businesses such as Lyft, Tinder and Redfin as well as enterprises such as Comcast, Under Armour, BMW, Nordstrom and Toyota depend on DynamoDB's scale and performance to support their mission-critical workloads.

DynamoDB is used by Lyft to store GPS locations for all their rides, Tinder to store millions of user profiles and make billions of matches, Redfin to scale to millions of users and manage data for hundreds of millions of properties, Comcast to power their XFINITY X1 video service running on more than 20 million devices, BMW to run its car-as-a-sensor service that can scale up and down by two orders of magnitude within 24 hours, Nordstrom for their recommendations engine reducing processing time from 20 minutes to a few seconds, Under Armour to support its connected fitness community of 200 million users, Toyota Racing to make real time decisions on pit-stops, tire changes, and race strategy, and another 100,000+ AWS customers for a wide variety of high-scale, high-performance use cases.

With all the real-world customer use, DynamoDB has proven itself on those original design dimensions:

- **Scalable** – DynamoDB supports customers with single tables that serve millions of requests per second, store hundreds of terabytes, or contain over 1 trillion items of data. In support of Amazon Prime Day 2017, the biggest day in Amazon retail history, DynamoDB served over 12.9 million requests per second. DynamoDB operates in all AWS regions (16 geographic regions now with announced plans for six more Regions in Bahrain, China, France, Hong Kong, Sweden), so you can have a scalable database in the geographic region you need.
- **Secure** – DynamoDB provides fine-grained access control at the table, item, and attribute level, integrated with AWS Identity and Access Management. VPC Endpoints give you the ability to control whether network traffic between your application and DynamoDB traverses the public Internet or stays within your virtual private cloud. Integration with AWS CloudWatch, AWS CloudTrail, and AWS Config enables support for monitoring, audit, and configuration management. SOC, PCI, ISO, FedRAMP, HIPAA BAA, and DoD Impact Level 4 certifications allows customers to meet a wide range of compliance standards.
- **Durable and Highly-Available** – DynamoDB maintains data durability and 99.99 percent availability in the event of a server, a rack of servers, or an Availability Zone failure. DynamoDB automatically re-distributes your data to healthy servers to ensure there are always multiple replicas of your data without you needing to intervene.
- **Performant** – DynamoDB consistently delivers single-digit millisecond latencies even as your traffic volume increases. In addition, DynamoDB Accelerator (DAX) a fully managed, highly available, in-memory cache further speeds up DynamoDB response times from milliseconds to microseconds and can continue to do so at millions of requests per second.
- **Manageable** – DynamoDB eliminates the need for manual capacity planning, provisioning, monitoring of servers, software upgrades, applying security patches, scaling infrastructure,

Recent Entries

From Schooling to Space: Eight Predictions on How Technology Will Continue to Change Our Lives in the Coming Year

Strengthening our commitment to Australia

How the Seahawks are using a data lake to improve their game

Understanding Climate Change Using High Performance Computing and Machine Learning

Grüezi Schwiiz! Bonjour la Suisse! Buongiorno Svizzera! An AWS Region comes to Switzerland

Reinventing virtualization with the AWS Nitro System

Now Go Build - Season 2

The OS Classics

The journey to modern manufacturing with AWS

Reaffirming our commitment to Italy: Introducing the AWS Europe (Milan) Region

Confermando il nostro impegno in Italia: Apertura della Regione AWS Europe (Milano)

Continuing our investment in Africa: Introducing the AWS Africa (Cape Town) Region

When scaling your workload is a matter of saving lives

How Amazon is solving big-data challenges with data lakes

The power of relationships in data

AWS's HPC leadership recognized by industry experts with HPCwire awards

¡Hola España! An AWS Region is coming to Spain!

Act locally, connect globally with IoT and edge computing

Modern applications at AWS

Expanding the cloud to the Middle East: Introducing the AWS Middle East (Bahrain) Region

monitoring, performance tuning, replication across distributed datacenters for high availability, and replication across new nodes for data durability. All of this is done for you automatically and with zero downtime so that you can focus on your customers, your applications, and your business.

- **Adaptive Capacity** –DynamoDB intelligently adapts to your table's unique storage needs, by scaling your table storage *up* by horizontally partitioning them across many servers, or *down* with Time To Live (TTL) that deletes items that you marked to expire. DynamoDB provides Auto Scaling, which automatically adapts your table throughput up or down in response to actual traffic to your tables and indexes. Auto Scaling is on by default for all new tables and indexes.

Ten years ago, we never would have imagined the lasting impact our efforts on Dynamo would have. What started out as an exercise in solving our own needs in a customer obsessed way, turned into a catalyst for a broader industry movement towards non-relational databases, and ultimately, an enabler for a new class of internet-scale applications.

As we say at AWS, It is still Day One for DynamoDB. We believe we are in the midst of a transformative period for databases, and the adoption of purpose-built databases like DynamoDB is only getting started. We expect that the next ten years will see even more innovation in databases than the last ten. I know the team is working on some exciting new things for DynamoDB – I can't wait to share them with you over the upcoming months.

Comments

Graeme Roberts • 3 years ago
Much more of a miracle to me than any of the artificial intelligence applications to date.
7 • Reply • Share ›

Todd ↗ Graeme Roberts • 3 years ago
I am not sure about that, because anytime you can take the human element out of the equation while enhancing the system to make decisions on the fly from different data-sources and types is like the discovery of electricity. This tells me that we are moving in a direction that is unprecedented.
• Reply • Share ›

assignment writing help • 3 years ago
Good internet-scale application which is really good work. It is an application which is

done everything, i want this types of apps for working different types of work. Thanks a lot.

Reply • Share ›

Thomas Ray • 3 years ago

Werner, Its really the best post I have read till date about Dynamo DB. You have detailed it in such perfection that no body does. Keep up the good work. I always follow your post as I am also writing about **data integrity** niche.

Reply • Share ›

Neil Anderson • 3 years ago

Werner, No doubt this is is the best the best DynamoDB intro/overview so far. Great article thanks and keep it up!

Reply • Share ›

Jason • 3 years ago

Here's to hoping future Dynamo libraries in the AWS SDK for Node.js support optional async methods to make waiting on queries to return much easier on devs!

Reply • Share ›

UtterlyCloud • 3 years ago

That was really nice, I've created a website based on cloud, hosted on cloud. Please visit www.utterlycloud.com

Reply • Share ›

Todd ➚ UtterlyCloud • 3 years ago

I looked at the site, I love the concept, but there are a few things I would change:

→ The frame on the left needs to be moved over so I can we see the wording
→ "is the onethe best ", maybe change that to "is one of the best CSPs"
→ AWS - to keep consistency, include the date AWS was made available to the public

I do like the step-by-step procedures to help individuals create cloud resources, very nice.

Todd

Reply • Share ›

UtterlyCloud ➚ Todd • 3 years ago

https://www.utterlycloud.co... .Your feedback is really helpful to us.The above contains a article on basic understanding of DynamoDB for beginners. Have a look at it and provide feedback. We have number of articles on other cloud services. Please go through them too. We are going to push more articles and tutorials in coming days. Please go through them and please share with your friends who want to learn cloud. People like you really motivates us. Thank you so much.

Reply • Share ›

William Cox • 3 years ago

If DynamoDB is still "day one" then how come the Release Notes show no changes have been made for nearly 2 years? Nov. 2015 is the last change noted: https://aws.amazon.com/rele...

Reply • Share ›

Prashant Singh • 3 years ago

I have one humble question. If Hashing is used to distribute the keys over the ring, and if hash function is any good, why do Dynamo had to employ vnode concept?

Reply • Share ›

Sreedhar Kukunooru ➚ Prashant Singh • 3 years ago

Prashant, Refer to this white paper (section 4.2). That should answer your question.

http://www.allthingsdistrib...

Reply • Share ›

Sean Anderson • 3 years ago

By far, the best DynamoDB intro/overview I have ever seen.

Reply • Share ›

Luis Trigueiros • 3 years ago

Thank you found it very interesting read.

Reply • Share ›

joono • 3 years ago

Pretty interesting indeed

Reply • Share ›

Avraam J. Dectis • 3 years ago

.
That was actually kind of interesting.
.

Reply • Share ›