redis

Latency is the New Outage

WHY SPEED IS THE NEW TABLE STAKES

Executive Summary

As organizations increasingly leverage applications hosted on different infrastructure, in different geographies, and with different providers, the speed at which end users can access those applications comes under pressure.

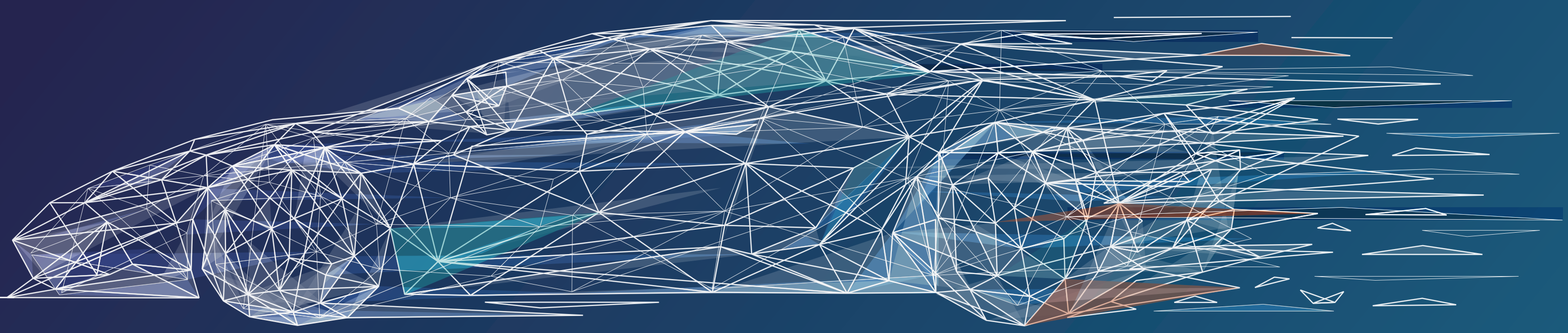
Add to this the fact that applications today are made up of significant numbers of different component parts, and you have a recipe for a degraded user experience.

These dual traits—application modularity and infrastructure complexity—can directly result in poor application performance.

It is for this reason that the speed of the data layer, the common horizontal layer across the application, is critical.

Being able to use a geographically replicated data layer, while avoiding the issues around data inconsistency, is a challenge that all IT leaders need to resolve.

By leveraging a data layer that unifies your data across clouds and the globe, organizations can overcome some of the inherent limitations that have challenged technology teams for decades and deliver better experiences to their end users.



Introduction

Digital teams have spent the past decade ensuring their digital assets are available at all times, and they have largely succeeded! High availability is now the norm.

Organizations have, in part, achieved this high level of digitalization and high availability by leveraging the benefits the cloud brings—ease of scalability, modular services, more refined architectural patterns. These characteristics all enable positive outcomes, but do so with the flip side of increased complexity. That complexity was initially most impactful in terms of availability and gave rise to what we call the Availability Epoch. However, as organizations better understand how to deliver high availability, they find there are still other problems to solve.

But now the Availability Epoch is beginning to wane, as organizations increasingly look to reduce latency as the next key to unlocking the outcomes they seek. They increasingly understand that slow products and services might as well not be available at all—that latency is the new outage.

Unfortunately, resolving latency issues is often harder than creating high availability. While availability can be improved through good engineering, greater levels of redundancy, and better monitoring and visibility, the latency conundrum is constrained by the very laws of physics.

To reduce latency as much as possible, organizations need to understand what latency is and the factors that contribute to it, and have clear, definitive guidelines for reducing, as much as possible, latency for the users of their applications and websites.

If latency is the new outage, here is the intelligence you need to deliver the lowest latency physically possible.

Organizations increasingly understand that slow products and services might as well not be available at all—that latency is the new outage.

It's not the big that eat the small, it's the fast that eat the slow

Moving fast is the new normal. While analysis and prudence were once the name of the game, the operating reality today is that to stay ahead of their competitors, organizations need to innovate faster than ever before. The operational landscape for every organization is changing rapidly and success will be granted to those who can best react to this dynamism.

WHY AGILITY IS SO IMPORTANT: TIME IS OF THE ESSENCE

The world we live in today is markedly different from that of only a few short years ago, and the rate of change continues to increase. Within this context, giving organizations the ability to move fast is more important than ever before. It is important to understand the changes that are occurring in society to better understand just how important moving fast really is.

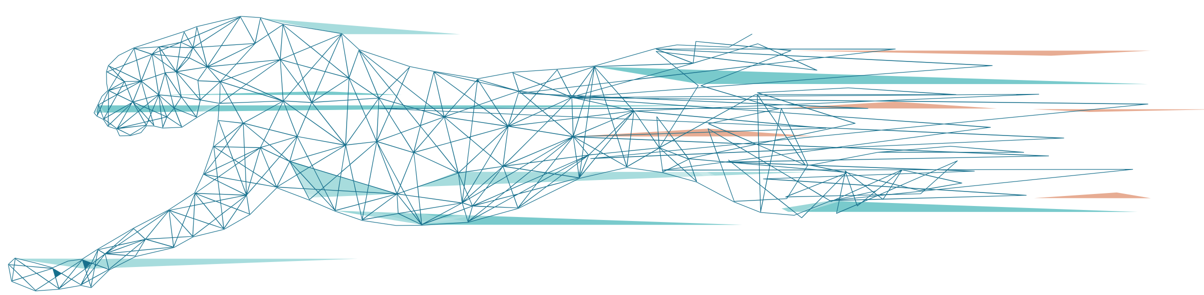
Back in 2011, noted entrepreneur, investor, and board member Marc Andreessen (the inventor of the Netscape web browser) wrote a now-famous opinion piece for ***The Wall Street Journal***, explaining Why Software Is Eating the World.

In his essay, Andreessen put forward his theory about the size, scope, and speed of this change, suggesting that: "we are in the middle of a dramatic and broad technological and economic shift in which software companies are poised to take over large swathes of the economy."

But while the fact that this change is of great importance, it is the speed of this change which is of most relevance here. Fundamental to the ability to move fast is the ability to make changes, leverage a variety of tools, and deliver the best end-user experience. If all this sounds very much like the way that Silicon Valley companies work, that makes sense. The fact is that many of the organizations seeing success disrupting traditional industries are looking and feeling more and more like entrepreneurial technology companies.

It was almost ten years ago that Andreessen wrote that essay, and many of his predictions have come to pass. While it has become a cliché to use Tesla, Uber, Lyft, Netflix, and Airbnb as examples of digital disruption, it's safe to say that executives in both taxi and hospitality companies have been hit by a tidal wave of unprecedented proportions. Beyond the cliché, however, the important thing to note is just how much effort these companies go to in order to deliver the fastest possible customer experience on their applications: speed really does matter.

It is a truism that moving fast in an organizational setting is predicated on providing digital experiences which themselves display these attributes of speed. Latency is the new blocker of organizational transformation.



Changing the world with a move to digital

A massive number of examples of traditional organizations are pinning their hopes of future success on a move to digital. It is worth looking at a few examples to get a sense of this scale.

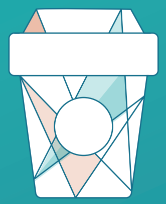
DIGITAL JOE: STARBUCKS GOES DIGITAL-FIRST

Starbucks CEO Kevin Johnson was once an executive at Microsoft. His experience at the giant technology company—also based in the Seattle area—helped him apply digital thinking to his new role at a very different sort of organization.

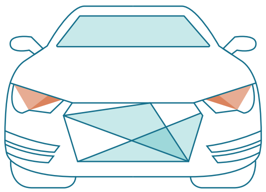
Johnson talks about Starbucks' digital journey plainly: "Where others are attempting to build a mobile app, Starbucks has built an end-to-end consumer platform anchored around loyalty."

The company's main digital innovation centers around its Mobile Order and Pay app. Focusing on the app is fundamentally a customer-first strategy, as it addresses the basic wants of the consumer: convenience, line avoidance and fulfillment speed, and so forth. Coupled with its extensive loyalty program, the app gives Starbucks the perfect venue to upsell and market to consumers. Just as important, the app funnels back massive amounts of user data to the company, allowing it to better understand its customers' habits and desires.

Starbucks invested heavily in creating digital touchpoints for its customers and with its massive global footprint, application availability—in terms of both raw uptime and latency—was critical.



Focusing on the app is fundamentally a customer-first strategy, as it addresses the basic wants of the consumer.

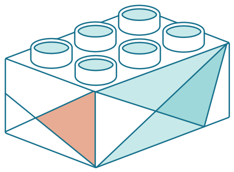


AUDI: AUTOMAKER OR DIGITAL CORPORATION?

The already highly competitive automotive industry faces massive disruptive force in the short- to mid-term. New sales models, the rise of electric vehicles, and autonomous driving are all changing the game for car makers. Faced with these challenges, Audi has changed how its vehicles are sold.

Launched in 2012, Audi City provides a deep brand experience that enables visitors to virtually explore the entire range of Audi products, even in city-center stores without enough room for showrooms.

Audi is a luxury brand and the company's move to disrupt its own sales channel was not taken lightly. Audi invested highly in building a virtual retail experience that was as authentic as a physical one. Part of this process included the utilization of various different touchpoints, application form factors and display approaches. Doing all of this within users' expectations of a speedy experience was a technology stretch that required new thinking.



LEGO: FROM PLASTIC BRICKS TO DIGITAL BLOCKS

The LEGO Group is the famed Danish manufacturer of the eponymously named children's toys. But after a long period of expansion from 1970 to 1991, LEGO suffered a steady decline in its business from 1992 to 2004. By 2004 the company found itself close to bankruptcy.

Reaching a tipping point, LEGO was forced to begin a major restructuring. Its digital transformation focused on nurturing new revenue sources coming from movies, mobile games, and mobile applications.

As LEGO embarked upon this process, one of the key limitations it had to overcome was the performance impact of tens of thousands of kids simultaneously using their various LEGO apps and games. Management dictated that speed—of innovation and of the delivery of its digital products—was a non-negotiable requirement.

LEGO Management dictated that speed—of innovation and of the delivery of its digital products—was a non-negotiable requirement.

The two epochs of digital delivery

Organizations have experienced two epochs when it comes to digital delivery. First they had to contend with the Availability Epoch. Today, as availability becomes largely a solved problem, they're entering the Speed Epoch.

THE AVAILABILITY EPOCH: UPTIME IS KEY

With the advent of the internet and creation of companies like Amazon, eBay, and Netflix, corporations began to explore the potential of these new technologies and business models. In the early days of digital transformation, IT teams primarily chased a singular metric: uptime. Organizations moving into the digital world had one focus: ensuring their websites and applications would be available anywhere, anytime. This time, which we call the Availability Epoch, was characterized by tools and approaches that ensured site reliability.

The Availability Epoch fostered a huge amount of innovation, all in an effort to increase the number of 9s in the uptime percentage metric. The moulding of the development and operations function into the combined DevOps role was intended to speed application development and increase reliability. Powerful application and infrastructure monitoring tools and platforms were created in order to achieve this holy grail: ever-higher uptime percentages in a more rapidly evolving environment.

Indeed, while the "five-9s" goal rolls off the tongue, it is important to understand what 99.999% uptime actually means: no more than a mere 26 seconds of downtime per month. As more and more companies approach or achieve uptime statistics like this via high-quality engineering and a deep understanding of what it takes to plan for failure, CIOs have been able to focus on other areas for improvement. Consequently, areas that were once ignored are now becoming critical.

As availability becomes largely a solved problem, organizations are entering the Speed Epoch.

THE STATISTICS THAT POINT TO THE END OF THE AVAILABILITY EPOCH

Organizations have spent the last decade or two being told that, as they move to more digital touchpoints with customers, the fundamental availability of those touchpoints is key. An entire generation of IT practitioners have been obsessed with availability metrics and tools to improve them.

There are, however, some fundamental factors changing the game for these practitioners. Aside from the increased complexity that their own efforts to engineer for uptime have created, there are also external factors that drive critical requirements for the lowest latency possible.

As consumers move en masse to mobile touchpoints, the very way they consume data, and their requirements for immediacy, are changing. Consumers are using their mobile devices to better inform themselves about the products and services that are important to them. 80% of consumers look up product information, reviews, and prices on their smartphones while shopping in a physical store.

And this tendency to consume information is just the start, consumers are also transacting in new ways. Fully one-third of all purchases during the 2018 holiday shopping season were made on smartphones.

Unfortunately, organizations have a tendency to overestimate their own ability to deliver good experiences. Qualtrics research found that while 60% of companies think they're providing a good mobile experience, only 22% of consumers feel the same.

All of which points to this need for speed—mobile browsing happens in different contexts from fixed browsing, while walking, in-store, and in short breaks—all of these contexts demand speed more than ever before.

A CAUTIONARY TALE: DISNEY'S DISMAL LAUNCH

Last year, Disney staked much of its future success on the roll out of Disney+, the company's high-profile streaming video service. Like many organizations looking to make a huge splash in an area outside its usual delivery method, Disney talked up the launch and hyped customers for the experience they were about to see.

Unfortunately, as soon as Disney+ launched, customers began to complain about the service's poor performance: extended buffering, dropouts, and general latency all hampered what could have been a jubilant launch day. The criticism was clear: a service delivering poor speed is just as bad as one that is entirely unavailable.

THE SPEED EPOCH: SNOOZE AND YOU LOSE

Over the past handful of years, most companies have gained a good understanding of uptime. Meanwhile, their service providers have done much to bake multiple redundancies into their platforms, ensuring that the path to almost perfect availability is easy to navigate. Monitoring tools, site reliability engineering practices, and the embrace of resilience in the event of inevitable failures have helped deliver what end users now expect: websites and applications that are available whenever they're needed.

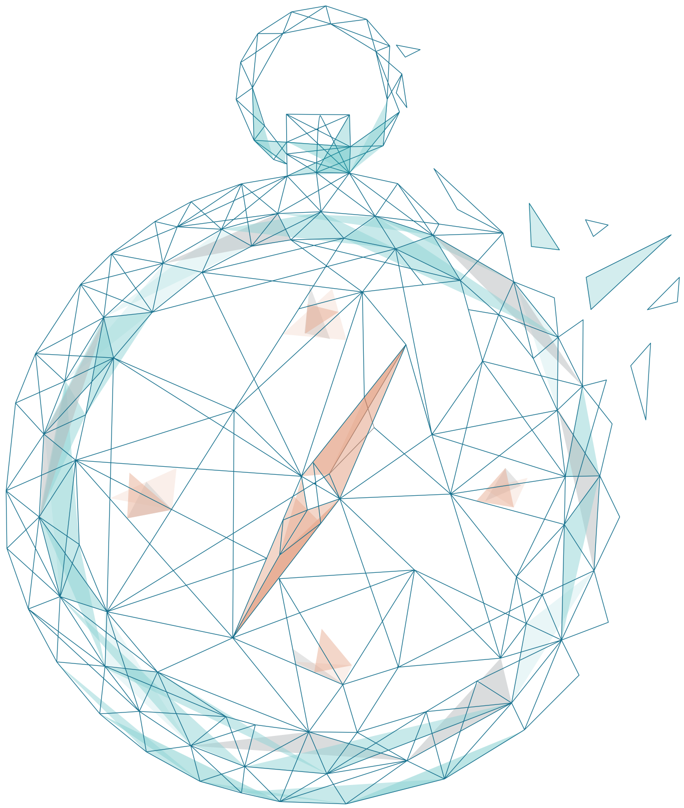
But all of this extra engineering, and the leveraging of ever more complex architectures in an effort to deliver the most resilient applications, has introduced new challenges, which are just as critical as uptime.

Clearly, we're entering a second epoch, one in which reliability has become table stakes while speed is now the competitive differentiator. Customer decisions, formerly made with time and analysis, are increasingly made in a heartbeat. And if your site takes more than that heartbeat to load, or your streaming service suffers from stalls and buffer-pauses, you're set to lose.

Lest you assume that customer dissatisfaction doesn't impact upon their consumption habits, think again. As detailed in a 2019 Forbes article (How Fast Is Fast Enough? Mobile Load Times Drive Customer Experience and Impact Sales): "A slow loading page on a mobile device doesn't just try consumers' patience. It can be the customer experience 'fail' that costs you a sale. This is the key takeaway of the 2019 Page Speed Report... The study, which explores the attitudes of 1,150 consumers and businesses, finds that page speed is a deciding factor in purchasing behavior."

And the impact of poor page speed is not inconsequential: "nearly 70% of consumers say page speed impacts their willingness to buy. What's more, a slow loading time also lessens chances they will return in the future. A breakdown of the data reveals 22% of shoppers said they would close the tab and 15% said they would visit a competitor's site and 12% would tell a friend about their negative experience."

If the new epoch is defined by the need to ensure latency is low as possible, what are the things organizations need to think about to reach that goal?



Delivering speed in a complex world

In his seminal 2013 post on The Composable Enterprise™, Jonathan Murray, former CTO of Warner Music Group, described the future of technology within the context of enterprise demands for speed and agility. Based on his lifetime's experience delivering upon large corporations digital strategies, Murray described the Composable Enterprise this way: "Business functions, processes, organizations, supplier relationships, and technology need to be seen as building blocks that can be re–configured as needed to address the changing competitive landscape."

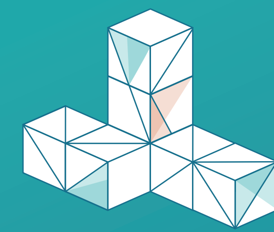
This new Component Operating Model (COM) requires a 'Lego brick' approach to designing and implementing processes and the organizations that support them. Implementing a COM based approach will have profound impacts on the structure of organizations, the nature of work.

Business designs based on COM will create significant stress for traditional IT infrastructures and organizations. Our current IT services were built to serve a static—and often functionally siloed—operating model. IT needs to become much more dynamically adaptable to keep pace with the speed of business today.

"A new Component Architecture Model (CAM) approach to IT infrastructure, applications, and services will be required to ensure that IT can deliver what the business needs. The time between identifying a business need and delivering the required IT solution needs to becomes hours and days rather than months and years."

Written several years ago, Murray's prescient post describes the new normal situation within organizations. We have seen, over the past handful of years, a seismic shift in the way infrastructure is used, and applications are built. With the rise of containers, microservices architectures, discrete modular application tools, and the like, keeping an application functioning and ensuring it functions well means juggling dozens of services, regions, geographies, service providers and more.

So while all of this composability drives developer productivity and organizational agility, it comes at a cost. It would seem that delivering low latency under these conditions is a pipe dream.



We have seen, over the past handful of years, a seismic shift in the way infrastructure is used, and applications are built.

Speedy data in a distributed computing environment

As we have seen from Murray's seminal work on composability in modern applications and infrastructures, we no longer have a simple monolithic stack upon which applications are built. Rather, in an effort to give developers and their organizations the most flexibility and highest velocity possible, we leverage a huge number of modular developer services, different infrastructure patterns, various hosting approaches and massively distributed geographical spread of applications. All the while, we're trying to deliver these applications as fast as possible to users all across the globe.

In this time of massive complexity, it would be easy to think that there is no common fabric that organizations can rely on—their world seems perpetually fluid and ever changing.

There is a common thread, however, that weaves its way through all the different things the organization does—and that is data. By thinking about a data layer as a consistent and unified thread leveraged by all the other parts of the stack, we enable organizations to make sense of the chaos. And by choosing a data layer that is engineered for distributed environments, that displays the very fastest processing times, and that delivers best-in-class resilience, we can deliver exactly what an enterprise needs.

A key way organizations can ensure their applications are both resilient and fast is to work within a consistent data layer construct. And consistent data begins with a database that can deliver seemingly impossible goals: distributed architectures, consistency, flexibility, and speed.

Modern approaches to reduce latency

As we have seen, applications are increasingly built using microservices: leveraging a multitude of different component parts, with different approaches to infrastructure, hosted in a variety of different locations, consumed by people everywhere, and distributed on many different platforms.

With data located in so many places and transmitted across so many different networks, it is hardly surprising that there are plenty of opportunities for data conflicts to occur. To deal with these conflicts, [conflict-free replicated data types (CRDTs)] were developed to let data be replicated across multiple locations.

With CRDTs, individual replicas can be updated independently and concurrently without any attendant coordination between them. Without CRDTs, concurrent updates to multiple replicas of the same data, without coordination between the computers hosting the replicas, can result in inconsistencies between the replicas.

With CRDTs, however, any inconsistencies that result from this distributed approach can be resolved. CRDTs were initially used in situations where mass distribution is the norm—online chat systems, internet gambling and audio and video streaming—but increasingly see usage in more generic applications.

There is significant underlying technology that goes into making a CRDT work, but the simplest way to think of it is that a CRDT provides a data layer whereby replicas can act autonomously and still provide consistency.

With data located in so many places and transmitted across so many different networks, it is hardly surprising that there are plenty of opportunities for data conflicts to occur.

Moving on from cache

On traditional database models, the database site is separate from the cache. Think of the database as the main municipal library and the cache as the local branch library, where the most popular books are kept to fulfil the most common borrower demands. If the most popular books are consistent, that might work OK, but as reading habits change, and new books come into, and go out of, favor, that becomes more difficult.

And this notion of rapidly checking different pieces of information on a constant basis is just the metaphor for modern applications—all of the composability that Murray talked about results in data having to be accessed from the database from many different services and places, and at many different times.

In a world with ever more discreet services and, hence, ever more places that something can go wrong, the traditional model isn't ideal (see diagram below). And in applications where data models are more about transferring many bite-sized chunks of information, the cache model may not be the fastest way to get data where it needs to go.

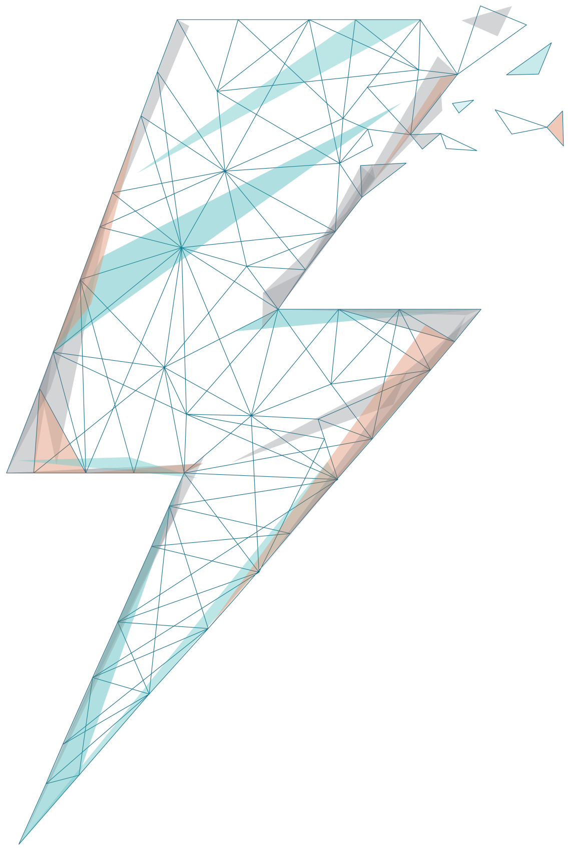
This is where the notion of a single data layer comes in—by leveraging a single layer to replace the database/cache combination, complexity of the data layer is reduced. In return, what is turbocharged is the distributed and modular application that is the norm today.

The added advantage is that reducing the number of parts at the data layer also reduces latency. While individual parts of a complex data layer may be fast, having a single data store reduces the number the network hops that invariably slows things down.

In place of a cache, therefore, many modern databases leverage in-memory techniques where memory, rather than external disks, is used for storage. This is critical since with everything stored in-memory, speed isn't constrained by multiple storage layers. With a cache-based model, what's stored in the cache becomes the bottleneck that limits overall speed.

In a traditional model, accessing data looks means the application has to:

Speed: One byte at a time



Traditional databases, as we have seen above, rely on external memory for their cache. Until very recently and for the past 50+ years, storage happened on physical spinning disks and hence most traditional database approaches were optimized for this.

But, since hard drives are physical devices, they have constraints that the physical world creates. To get around these physical constraints, a number of operating constraints were created. While a technical detour, the minutiae of physical disc technology impacts greatly on database speed.

The crux of the matter, however, is that the line between modern storage and memory is becoming less distinct. The rise of solid state drives (SSDs) and other new storage approaches mean that these engineering work-arounds, designed for a world constrained by the physical speed of mechanical devices, are no longer required. It also means that storage can be tiered, such that all data can be kept in fast storage and the need for a separate cache ends.

The net result, for those trying to engineer for speed, is a faster data layer upon which to build our applications.

The rise of new storage approaches mean that engineering work-arounds, designed for a world constrained by the physical speed of mechanical devices, are no longer required.

Scaling without hampering speed

It's all very well to create an application that runs fast with limited usage, but what happens when your throughput scales hugely? This is the problem that every application developer, looking for uptake and virality, hopes to face.

But scaling happens in two ways—***up***, in terms of how much data is transferring across the data layer, but also ***out*** in terms of the sheer amount of information that exists.

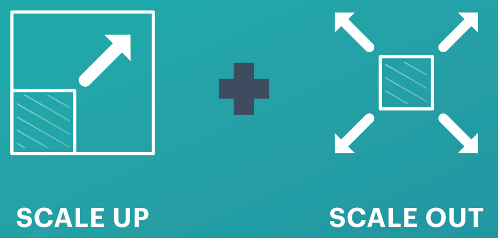
Organizations need to build a data layer that allows for this scaling in a staged and seamless way. This entails thinking about a number of different factors: the ability to run the data layer in multiple locations, the ability to use different types of memory and storage, the ability to tier data depending on its regularity of usage, and finally, the ability to scale globally.

Let's turn to this last area. All of this ability to store and process in-memory is good, but if your application needs to spread around the world, can you still enjoy this same low level of latency?

IT'S A MULTI-CORE WORLD

Modern processing increasingly happens with a multi-core contact. Multi-core simply refers to computing where two or more individual processing units exist within one CPU. The instructions being sent to the CPU can be processed on separate cores at the same time, increasing overall speed.

Leveraging multi-core architectures can be challenging. Organizations wishing to leverage a data layer that can scale in the most performant way need to think about this. Is your data layer able to scale out on a single cluster to deliver the best scale with the lowest latency?



Organizations need to build a data layer that allows for this scaling in a staged and seamless way.

A WALK DOWN CAP LANE

Since this paper will inevitably be used by those aspiring to build globally distributed applications that display performance akin to locally located ones, it is worth looking at some constraints around distributed data layers.

Some 20 years ago, computer scientist Eric Brewer developed the CAP theorem, which relates to distributed applications and, specifically, the data those applications create and consume.

The CAP theorem, in the simplest terms, asserts that any networked shared-data system can have only two of three desirable properties; consistency (C) equivalent to having a single up-to-date copy of the data; high availability (A) of that data (for updates); and tolerance to network partitions (P).

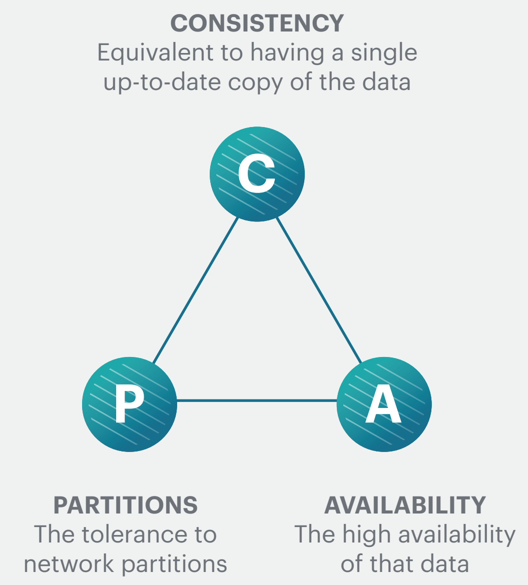
And given that, in these early days where a quest for speed was all, the CAP theorem meant that the approaches most likely to give the fastest speeds and application availability (network partitions and high availability) would also result in data inconsistency.

In the decades since the CAP theorem's introduction, however, new approaches to handling distributed systems have been developed that allow that theoretically impossible feat: data consistency, availability, and partition-tolerance. The rise of new data approaches mean that we can have low latency without foregoing data consistency.

While this isn't the place for a definitive technical treatise, it is important for those with the responsibility for their organizations' application to understand the rudiments of how modern applications work.

As noted, in a world where applications are, by necessity, distributed, there will be multiple nodes included in many individual applications. In this multi-nodal situation, there are two broad options: **Active-Passive data** or **Active-Active data**.

THE CAP THEOREM



The rise of new data approaches mean that we can have low latency without foregoing data consistency.

UNIFIED DATA LAYER

Data planes, the part of the software that processes the data requests, can either be Active-Active or Active-Passive.

Active-Active (also sometimes called dual-active) is an approach whereby each node has access to a replicated database giving each node access and usage of a single application. This technology is what enables the ability to keep data consistent for your applications across different environments (servers, hybrid, multi-cloud) and even applications that are distributed across the globe. In an Active-Active system, all requests are load-balanced across all available processing capacity. Where a failure occurs on a node, another node in the network takes its place.

An Active-Active cluster is typically made up of at least two nodes, both actively running the same kind of service simultaneously. Because there are more nodes available to serve, there will also be a marked improvement in throughput and response times compared to an Active-Passive approach.

ACTIVE-PASSIVE

An Active-Passive cluster also consists of at least two nodes. However, as the name "Active-Passive" implies, not all nodes are active. In a cluster with two nodes, for example, if the first node is already active, the second node must be passive or on standby. The passive (a.k.a. failover) node serves as a backup that's ready to take over as soon as the active (a.k.a. primary) server gets disconnected or is unable to serve.

When clients connect to a two-node cluster in Active-Passive configuration, they connect to only one server. In other words, all clients connect to the same server. Like in the Active-Active configuration, it's important that the two servers have exactly the same settings. This is called redundancy and ensures that data can replicate seamlessly between the nodes.

If changes are made on the settings of the primary server, those changes must be cascaded to the failover server. So when the failover does take over, the clients won't be able to tell the difference.

If latency is the new outage, then clearly the closer a node is to the application user, the lower the latency figures will be. We therefore need to find a way to distribute applications globally (since distributing nodes close to application users reduces latency), while still ensuring consistency. Luckily we have some help in this regard.

BUILT WITH SPEED IN MIND

Conflict-free replication is a notion that allows multiple copies (replicas) of data to exist in multiple locations in a consistent manner. It is a very important method of ensuring low latency for distributed applications, but there are other aspects to consider. As noted above, modern databases designed to deliver the lowest-latency for modern applications store data in-memory. By removing the need for an external cache we can reduce the amount of data traffic required.

While traditional databases were designed for use cases where 10s or 100s of milliseconds processing time was acceptable, in today's world, where instant application response times are required, sub-millisecond performance is a necessity.

FAILING IS OK WHEN YOU DO IT FAST

Failover, as the name implies, is an automated system whereby, in the event that one node fails for some reason, another, replicated node picks up the slack. While failover is easy to engineer for, the speed of that failover is what determines the end-user impacts from the fail.

To ensure the lowest latency in a world in which nodal failures may be unavoidable, it is important that the multi-nodal data layer can deliver failover as quickly as possible.

Summary

In the modern world, organizations, driven to deliver digital experiences, need to ensure their stakeholders can use applications whenever and wherever they wish. But today's users demand not only continuous access but also virtually instant performance. Latency, in a world moving from the Availability Epoch to the Speed Epoch, can be as bad as application unavailability.

Luckily we have options today that simply weren't available a decade ago. Many impediments to delivering speedy applications, CAP Theorem among them, have been overcome. And now, organizations have the ability to leverage a data layer which is conflict free regardless of how many replicas are used.

By leveraging databases that work entirely in-memory, and running these in an Active-Active manner, we deliver faster databases than were previously available and deliver the low latency that today's application users demand.

This should be considered urgent business for every organization - your competitors and your disruptors are delivering fast applications and your customers are demanding them - you don't have the luxury of time.

About the Author – Ben Kepes

Ben Kepes is a technology analyst, commentator, and consultant and, over the past decade and a half, he has built up a significant following as a globally-recognized subject matter expert in the areas of cloud computing, enterprise technology, and digital transformation.

Ben's commentary has been widely published in such outlets as Forbes, Wired, and The Guardian and he has been invited to speak at a large range of technology, business, and general interest conferences.

@benkepes

About Redis

Modern businesses depend on the power of real-time data. With Redis, organizations deliver instant experiences in a highly reliable and scalable manner.

Redis is the world's most popular in-memory database, and commercial provider of Redis Enterprise, which delivers superior performance, matchless reliability, and unparalleled flexibility for personalization, machine learning, IoT, search, e-commerce, social, and metering solutions worldwide.

Redis, consistently ranked as a leader in top analyst reports on NoSQL, in-memory databases, operational databases, and database-as-a-service (DBaaS), is trusted by more than 7,400 enterprise customers, including five Fortune 10 companies, three of the four credit card issuers, three of the top five communication companies, three of the top five healthcare companies, six of the top eight technology companies, and four of the top seven retailers.

Redis Enterprise, available as a service in public and private clouds, as downloadable software, in containers, and for hybrid cloud/on-premises deployments, powers popular Redis use cases such as high-speed transactions, job and queue management, user session stores, real time data ingest, notifications, content caching, and time-series data.

Corporate Headquarters

700 E El Camino Real Suite 250
Mountain View, CA 94040

t: +1 (415) 930-9666
redis.com

Follow Us