For the last 30 years, a deluge of transformative trends has been continually remaking and accelerating the database industry. From a relatively staid corner of the mainframe world handling only carefully constructed data to modern microservices architectures working with a wide variety of unstructured data, database technology continues to morph and expand into new technologies, ever-increasing demands, and even more mission-critical use cases.

To fully understand what's happening in the database world right now—and what's coming next—it's important to work through the impact of these trends. Some of these trends have been shaping database design and implementation for decades, while others are brand-new forces with implications we can't yet fully fathom. Some stand alone, while others are deeply interconnected, born from and strongly tied to existing trends.
Instant performance

Modern databases have to respond in less than 1ms to avoid becoming a performance bottleneck.
In today’s digital economy, application performance is business performance. Modern customers demand instant response from the apps they use or they will instantly move on to a competitor.

But what does instant response really mean? Research indicates that apps have roughly 100ms—one-third of the time it takes to blink—before users begin to feel that they’re waiting for a response. The problem is that a request and response to the internet servers can easily take 50ms to make the round trip. The processing of the request can take another 50ms, which leaves no time for the database to respond.

That means the database has to be instant! That means modern databases have to respond in less than 1ms to avoid becoming a performance bottleneck.

Unfortunately, legacy databases running on spinning disks can take up to 100ms to respond, so engineers often build in a caching layer in DRAM to speed up common requests. But this adds complexity and can’t always serve every request quickly enough. Some modern databases are faster, but because they still take 10ms – 100ms to deliver results, they still need a cache.

To meet the needs of your customers and deliver truly “instant” performance, you need an in-memory database such as Redis, which can respond in less than 1ms, with no need for a separate caching layer (though Redis is still often used for caching on other databases).
Scalability

Linear scaling is critical to dealing with overall growth and seasonal surges without requiring massive infrastructure investments.
Growth happens! Modern apps are increasingly distributed, and you must plan for cost-effective ways to scale your data platform to meet overall growth—while also coping with seasonal surges and unexpected peaks in demand. Linear scaling of database performance is critical to making that possible without requiring massive infrastructure investments. While many cloud and on-premises databases claim the ability to scale linearly, they often struggle to prove it with solid benchmarks.

Redis Enterprise supports the open source Cluster API, enabling infinite, linear scaling by simply adding shards and nodes. In mid 2019, Redis Labs demonstrated Redis Enterprise extending its industry leading linear scalability with 200M ops/sec with less than 1ms latency on only 40 AWS instances.
Database designers are turning to tiered storage schemes, putting the hottest data in memory, while warm data resides on persistent memory.
Given the growing need for instant performance at scale, the rise of fast in-memory database architectures should not come as a surprise.

Almost all databases used to store data on spinning disks, as hard drives (HDDs) are typically the cheapest storage option. But spinning disks may not be fast enough to keep up with the demands of modern applications. That’s why database designers are finding ways to put even more data in faster DRAM—not just hot data that needs to be accessed constantly, but also “warm” data that requires occasional access—and even all the data.

Of course, even as memory costs continue to fall, DRAM remains more expensive than other storage options. So more and more database designers are turning to tiered storage schemes, putting the hottest data in memory, while warm data resides on persistent memory options like Flash and Intel Optane NVMe—which are faster than spinning disks but cheaper than DRAM. They’re getting 60% - 70% of DRAM performance at a much lower cost.

As an in-memory database, Redis is built to leverage DRAM’s superior performance. And Redis Labs has long been a leader in tiered-database storage, supporting Redis on Flash since 2016. In 2019, Redis Enterprise announced support for Intel Optane DC Persistent Memory, which delivers a new persistent-memory tier between DRAM and SSD.
High availability

The inability to quickly recover from a database failure can result in loss of data and millions of operations.
As organizations become more and more reliant on their applications for a wide variety of mission-critical use cases, those applications, and the databases behind them, have to be available at all times. No matter how reliable your systems and software, problems can occur. Obviously, downtime can reduce revenue, raise costs, and degrade the customer experience, but that’s only part of the problem. The inability to quickly recover from a database failure can result in loss of data and millions of operations, an unacceptable problem in use cases such as online orders or financial transactions.

Redis Enterprise offers **uninterrupted high availability** with five-nines (99.999%) uptime. Importantly, Redis Enterprise utilizes technologies such as diskless replication, instant failure detection, and single-digit-seconds failover across racks, zones, and geographies to deliver high availability using less resources than competing technologies. That kind of reliability can go a long way toward minimizing the operational burden on your IT teams.
Cloud native
(including hybrid-cloud and multi-cloud)

To maximize cloud benefits, every element of a modern technology stack has to work with a wide variety of resources delivered via the cloud.
In the beginning, you couldn't just buy a database, install it, and run it. You bought it and then hired an army of experts to install it, run it, and support it for you. You needed a crowd of skilled database administrators (DBAs) to make even small changes.

The rise of cloud computing has changed everything. With databases provisioned in the cloud, organizations no longer have to manage the internal workings of the databases they use. With a fully managed database service (a Database-as-a-Service, or DBaaS) delivered via the cloud, you get an endpoint to send and retrieve data from as needed and someone else deals with everything beyond that. The database is just a tool and far fewer people are required to manage huge workloads.

That's great, but taking full advantage of the cloud—being “cloud native”—is about more than just buying a DBaaS. To gain maximum cloud benefits, every element of a modern technology stack has to work with a wide variety of resources delivered via the cloud. To be truly cloud native, modern databases must play well with microservices architectures as well as containers and container orchestration systems such as Docker and Kubernetes.

And working well with a single cloud provider is only the beginning. Just as important, the database must be able to work across multi-cloud and hybrid cloud environments.

Many organizations leverage more than one cloud provider to mitigate vendor lock-in and gain access to particular capabilities that may not be available across the board. And even as almost all new workloads run in the cloud, many companies still prefer to keep some critical workloads on-premises.

Redis Labs was one of the first companies to offer zero-touch database provisioning. Today, Redis Enterprise Cloud comes in three different DBaaS versions: Essentials for development environments and low-throughput applications; Pro for high-throughput applications, many databases, or large datasets, and Ultimate, which offers custom plans optimized for any environment or use case.

To date, more than 2 billion Redis containers have been launched on Docker Hub, and a recent Sumo Logic survey found that Redis is the most-used database on Amazon Web Services, by far the industry’s largest cloud provider.
Simplicity and extensibility

Because no single data model can work well with all kinds of data, you can end up deploying multiple databases in the same organization, or even the same application—and nobody wants that!
Developers and operations teams increasingly seek operational simplicity in their databases—they want to use solutions that are easy and optimal for them and their particular use case. But because no single data model can work well with all kinds of data, they can end up deploying multiple databases in the same organization, or even the same application—and nobody wants that.

As a true multi-model database, Redis offers multiple data structures in the same database, cutting complexity and overhead by reducing the need to provision and maintain a different database for each data model required. Redis Enterprise incorporates 10 different data structures—including key-value, document, graph, and time-series data models—in a single, highly versatile multi-model database.

Much of that versatility comes from Redis modules like RediSearch, RedisGraph, RedisJSON, RedisTimeSeries, RedisBloom, and many others. Redis modules provide ultimate flexibility and extensibility to let Redis Enterprise effortlessly support probabilistic data structures, comprehensive search functionality, stream processing, deep learning, artificial intelligence, and more.

Developers know that Redis is extraordinarily easy for them to use, which is a key reason they love it. On Stack Overflow, Redis has been rated the #1 most-loved database for three years running. And that means there’s a huge number of experienced Redis developers with the skills required to create powerful new applications.
There’s increasing demand for a graphical user interface to manage the data layer and explore and interact with their data visually.
In addition to being simple and versatile, a modern database must also be easy for developers to use. As maximizing the value of data becomes even more important over time, it’s critical that a database platform support customer’s needs to massage and analyze that data quickly and easily.

While command line interfaces (CLIs) remain the most common way to work with a database platform, there’s increasing demand for a graphical user interface to make it easier to manage the system and support analysts who want to explore and interact with their data visually.

Redis Insight, announced last fall, is a free, consolidated tool combining a CLI and GUI to make it easier to write commands for all Redis data structures and create and use more optimal data structures and ways to to use them. Redis Insight also automates common bulk actions and helps analyze and reduce memory usage.
Open source DNA

Even the most conservative enterprises now realize there’s no reason not to rely on open source solutions for virtually any database use case.
It seems silly in 2020, but not that long ago open source software was not seen as up to the task of running mission-critical enterprise applications. OSS was virtually a four-letter word! Those days are long gone. Even the most conservative enterprises now realize there’s no reason not to rely on open source solutions for virtually any use case if it’s a good product with good support.

Today, the most innovative and powerful technologies are coming from open source roots. Having open source DNA is essential: only way to ensure a technology is as extensible as possible, and gets updated as quickly as possible. Even more important, perhaps, roots in open source means that developers address what the community really needs, rather than what a vendor can do or wants to do.

Redis Labs is the home of open source Redis, and developer of Redis Enterprise, built to deliver the best Redis experience. It maintains the simplicity and high performance of Redis, while adding many enterprise-grade capabilities.
NoSQL for the future

SQL databases will never go away, but NoSQL is the future—that’s where today’s most-interesting and most-valuable use cases are.
Traditionally, SQL databases modeled highly structured data in tables. With the rise of the internet and the explosion of new kinds of data, that was no longer good enough. Driven by internet services like Facebook and Twitter, most of the data in today’s databases is not structured, and would be very difficult to organize in a table. (You would have to use a matrix and would end up with tables that are 99% empty... very inefficient!)

To move beyond that, modern NoSQL databases put unstructured data in a document, using key-value data storage and strong search capabilities to find the right document and the right item in the document.

Does that mean SQL is dead? Of course not! SQL is still a major chunk of the database market. There will always be cases where SQL databases are the best way to order the data, and in case, it can take decades for legacy solutions to age out of the market. Increasingly, though, NoSQL is the future, that’s where today’s most-interesting/most-valuable use cases are coming.

Redis Labs has been named a leader in The Forrester WaveTM: Big Data NoSQL, Q1 2019. According to the report, Redis Labs customers “like its innovation for machine learning apps, performance, scale, customer support, and support or diverse NoSQL use cases.”