Building the Highway to Real-Time Financial Services

WHITE PAPER

redis

Executive Summary

Rapidly evolving customer behaviors and the accelerating speed of digital innovation pose extraordinary challenges for today's financial institutions. Successful institutions must build and deploy real-time services for use cases like online deposits, mobile transfers, electronic, payments, credit decisions, fraud detection, transaction scoring, and many more.

For decades, computing speed has been a competitive driver for traders seeking the fastest response to market changes. Increasingly, the fastest and most capable financial institutions attract the most valuable customers, but they'll also become the most valuable partners to other providers in the digital economy. While modern software architectures and the advent of cloud computing and containers have drastically reduced time to market and shortened development cycles, it is the evolution of the data layer that is enabling leaders in financial services to deliver instant personalization, real-time fraud detection, real-time transaction scoring, real-time investment decisions, and many other increasingly essential capabilities.

Companies constrained by legacy technologies particularly slow, traditional database architectures built for batch processing risk falling behind in the race to deliver real-time services. These legacy databases don't always scale gracefully and may not be fast enough, flexible enough, or reliable enough for real-time use cases. Successful financial institutions are rethinking their approach to the data layer by adopting database technologies that can provide the performance, real-time data models, and high availability needed to meet emerging real-time business demands. Redis Enterprise developed by Redis, the home of open source Redis is trusted by major financial services firms, including 3 of the 4 largest credit card issuers, to support a variety of real-time use cases, multi-cloud flexibility, and the advanced resilience and persistence financial services firms depend on.

> "Companies constrained by slow, traditional database architectures built for batch processing risk falling behind."

The technology trends driving the FinTech revolution

Businesses throughout the digital economy are reinventing their real-time capabilities using several key technologies:

Cloud computing
expands the capabilities of an enterprise with highly scalable, low-latency computing resources.

Serverless
execution allows cloud providers to dynamically provision resources as needed to manage large numbers of simultaneous connections.

Containerization
of applications enables faster development, simpler cross-platform deployment, and the ability to ensure that applications work exactly the same way wherever they're run.

Kubernetes
and other container orchestration solutions provide automated management of containers across cloud-computing providers, helping to avoid vendor lock-in by enabling multi-cloud deployment as well as easier integration of on-premises computing resources for hybrid-cloud architectures.

DevOps approaches
allow development teams to manage and implement rapid, resilient deployment of new features and services.

Microservices
and similar event-driven architectures enable developers to break down monolithic applications into collections of independent services, helping organizations boost the speed and agility of their software development and deployment.

Artificial intelligence
makes the best use of complex, disparate data sources to drive decision making in real time.

These technology trends are already reshaping large parts of the digital economy in sectors such as telecom, retail, transportation, and smart cities, and are now transforming financial services as well. Throughout the financial services industry, firms increasingly need to compete with real-time speed for use cases like customer onboarding, investment advice, credit decisions, customized offers, fraud analysis, and transaction scoring.

For decades, computing speed has been a competitive necessity in the quant-driven capital markets, in which traders seek every possible advantage in being the fastest to respond to market changes.

That's an extreme example, but real-time capabilities will also support competitive differentiation in numerous other areas of financial services. Not only will the fastest and most capable financial institutions attract the most valuable customers, but they'll also become the most valuable partners to other providers in the digital economy.

> “Throughout the financial services industry, firms increasingly need to compete with real-time speed for use cases like customer onboarding, investment analysis, credit decisions, customized offers, fraud analysis, and transaction scoring.”

To succeed in real-time business, financial institutions need the ability to process large datasets quickly enough to support instant response times across large user populations. Real-time financial institutions must be able to support :

Personalized
customer experiences across smart mobile and in-home devices.

Integration
with other providers for one-stop shopping for financial products and advice.

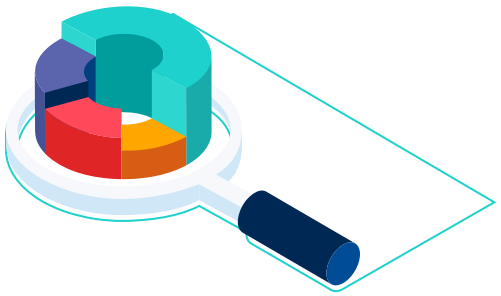
Fraud detection and mitigation
vs. global adversaries operating with real-time attacks.

Instant transaction scoring
and processing for risk management and regulatory needs.

AI and machine learning
support across application domains.

Data-driven investment
decisions aided by rapid analysis of large, heterogeneous datasets.

But how do financial institutions from legacy leaders to digital fintech disruptors and big-tech companies prepare for this real-time future?



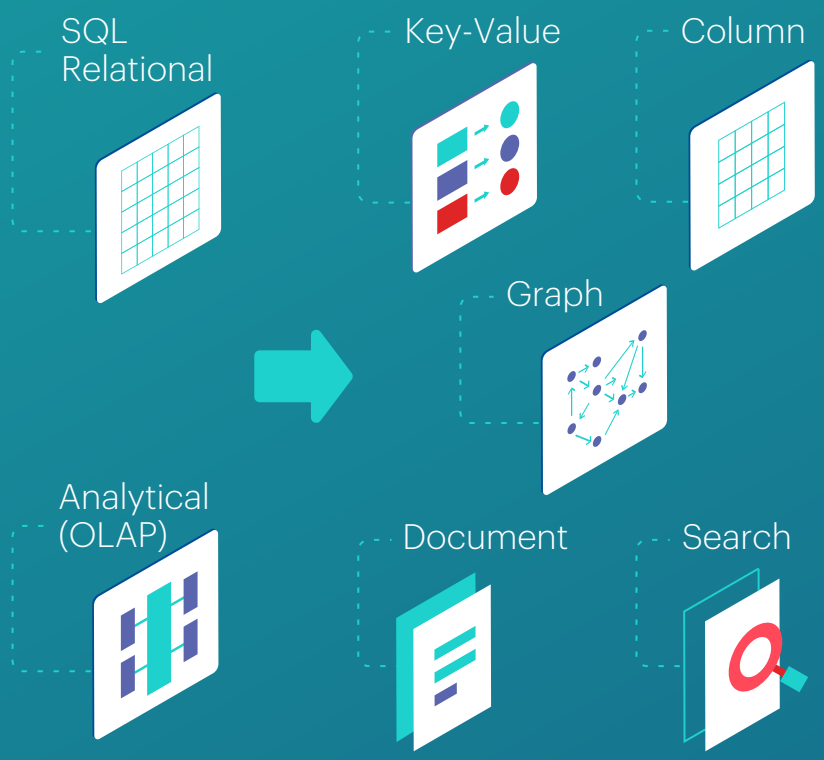
The real-time problem with legacy database architectures

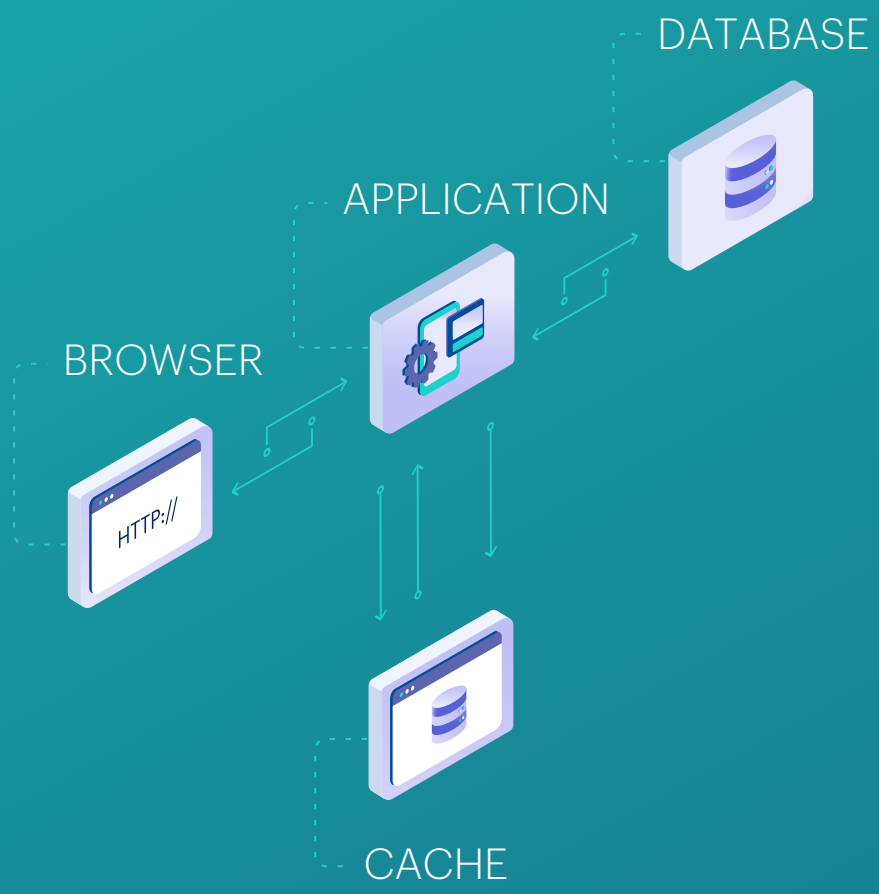
The financial services industry evolved as a batch-processing entity, with business processes built around the concepts of overnight processing runs, daily back-office reconciliation, and monthly statements. These processes were built using traditional relational database architectures with large volumes of data organized into rows and columns. These legacy database architectures still power much of the financial services industry today.

The financial services industry still has a place for time-tested business models such as bulk ACH payments with overnight settlements, 30-day credit-card billing cycles, and net settlement for securities transactions. Even so, the industry is increasingly embracing real-time payments, real-time e-commerce, and real-time gross settlement. Every batch process has its real-time equivalent, and financial institutions must support both at the same time.

Traditional relational databases are highly flexible tools. Given enough processing time, you can deploy an RDBMS for just about any application. Even if you don't want to jam up your production database with a monster query, it's easy enough to replicate the relevant data, pipe it into a separate data warehouse, and handle it from there.

> "The emergence of NoSQL database models gives software developers and architects more options for building the data layer. Selecting a database model optimized for the type of application data being processed has significant benefits for application. performance."





> Caching a slow database may add enough speed for certain use cases, but adds significant complications in the form of extra application logic, connections between multiple services, and multiple potential points of failure.

The problem is that you no longer have the luxury of time.

Disk-based database architectures were not built to support real-time applications. Although workarounds exist to extend disk-based databases with an in-memory cache, this patchwork approach contributes to higher complexity, higher coordination costs, scalability bottlenecks, and limitations in adopting the latest software architectures.

An in-memory cache may deliver enough speed for certain applications, but can get bogged down by excessive latency because not all data to be accessed is stored in the cache. Furthermore, you end up maintaining two separate environments, two separate code bases, and two separate sets of growth constraints.

At a certain point, "real-time" becomes harder to fake. In the rapidly approaching future of financial services, milliseconds matter. For example, what if you're trying to prevent fraud by conducting real-time analysis of behavior patterns across an entire ATM network? Or what if you need to enable ATMs with two-way video chat and built-in voice recognition, natural language processing, or automated language translation? Beyond the ATM, what if you have an opportunity to grab market share by participating in location-aware services that provide an augmented reality overlay based on personalized customer data?

Given the increasing maturity of easy-to-deploy technologies that support real-time capabilities, they are addressing an ever-widening variety of real-time use cases throughout the financial services industry. Proven real-time technologies already function well in many other contexts, and it's only a matter of time before the financial services industry takes even greater advantage of these technologies. People are becoming accustomed to real-time onboarding, real-time decisions, and instant service from digital financial providers, and customer expectations for everyone in the market are being transformed.

Legacy databases at financial institutions are already challenged to handle today's requirements. The unprecedented demands for large-scale, simultaneous access to real-time data goes far beyond what those databases were intended to support.

Legacy database drawbacks

In addition to their inability to support real-time applications, legacy databases have several other drawbacks:

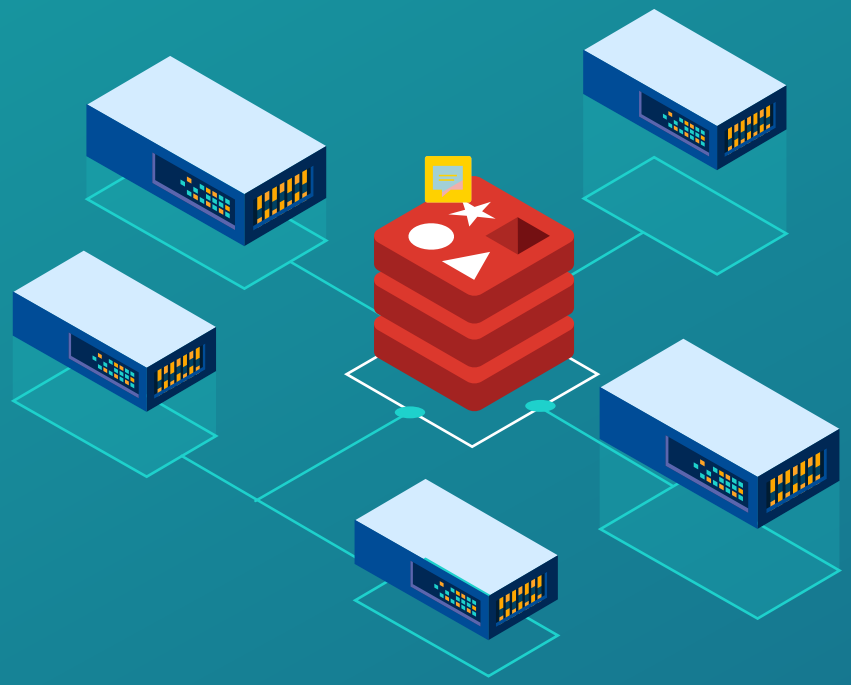
01. They don't scale gracefully.

Database administrators typically have to anticipate the expected size of an enterprise dataset, which drives specific choices around replication, clustering, backup and resilience. While that may have worked in the batch era given relatively stable numbers of customers, endpoints, and transactions, we are now operating in a much more fluid operating environment that involves highly variable data quantities, whether from social media posts, video feeds, location data, or IoT sensor data. Organizations need to scale their databases quickly and easily, without worrying about how long it's going to take to rebuild the indexes of a traditional database.

02. They're not flexible enough.

Legacy databases were built for SQL queries on relational tables. Yet today's most popular Internet services, including search and social media, are built using a diverse array of data models offering new ways of conceptualizing, representing, and using data. In some cases, legacy databases can be adapted for these new approaches, but there's a significant cost in terms of complexity and speed.

> "Legacy databases often have issues with scalability, flexibility, reliability, and complexity."

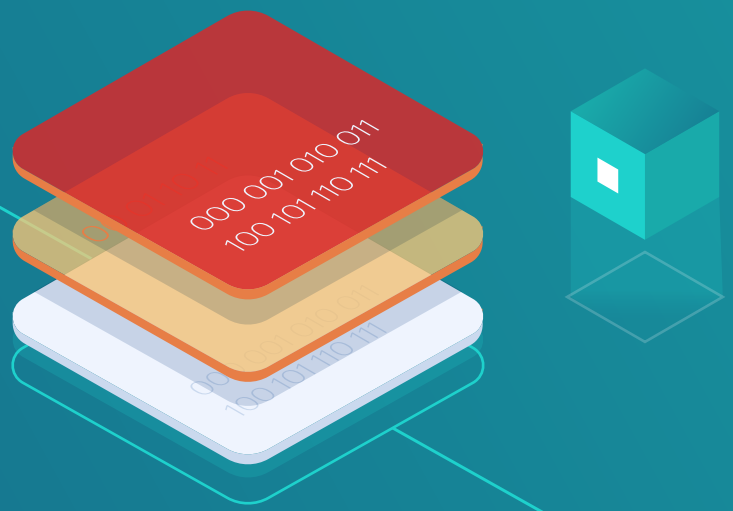


03. They're not reliable enough.

As the demands of the marketplace move toward real-time deliverables, financial institutions are encountering instances in which they have to recover from real-time system failure. In such instances, your resilience extends only as far as your persistence and high availability capabilities. The means to instantly recover without data loss and mitigate failures across multiple infrastructure regions is essential.

04. They're complex and difficult to work with.

Do you really want to force your developers to spend most of their time maintaining a multitude of complex legacy systems backed by unyielding and inflexible databases? To face the future with confidence, you need to keep your developers happy, innovative, and productive by empowering them to apply the latest techniques and data models to solve the most pressing challenges. That's the only way to ensure that your development teams are ready to respond rapidly to emerging business requirements.



> To make the transition to real-time financial services, you need to learn to work with the data layer in a new way.

Rebuilding the data layer for real-time

It's time to move beyond the batch era. Real-time financial services call for real-time capabilities that include high throughput, minimal latency, high scalability, high availability, and flexibility in data models.

To make the transition to real-time, you need to learn to work with the data layer in a new way. The shape of the data in your application matters to performance and capability which means that forcing everything into rows and columns is no longer sufficient.

Legacy approaches to database architecture are being superseded by a variety of better and faster ways to deliver upon these data-centric core competencies. Today's best-practice technology stack already powers flexible real-time services for the world's most popular applications in the digital economy, and these same technologies will inevitably transform financial services.

Digital fintech disruptors and technology companies have started to go down this path using the latest technologies. The good news is that there's still time for traditional financial institutions to follow suit. Now's the time to put your organization on a leadership footing by ensuring that your technology architecture anticipates marketplace demand for real-time data capabilities.

Becoming a real-time financial institution

What do you actually have to do in order to create tomorrow's real-time financial services? Every organization's needs are unique, but there are some best practices that can inform your efforts:

Caching is only the beginning.

Many companies have discovered that you can speed upyour existing database environment by adding an in-memory cache database to their legacy systems. This approach can be highly effective for getting a speed increase, as well as a practical way to get started with the technology. Yet this approach remains slower and more complex than working entirely within an in-memory database. If you're really committed to speed and simplicity, you need to take the next step and move entirely to an in-memory database.

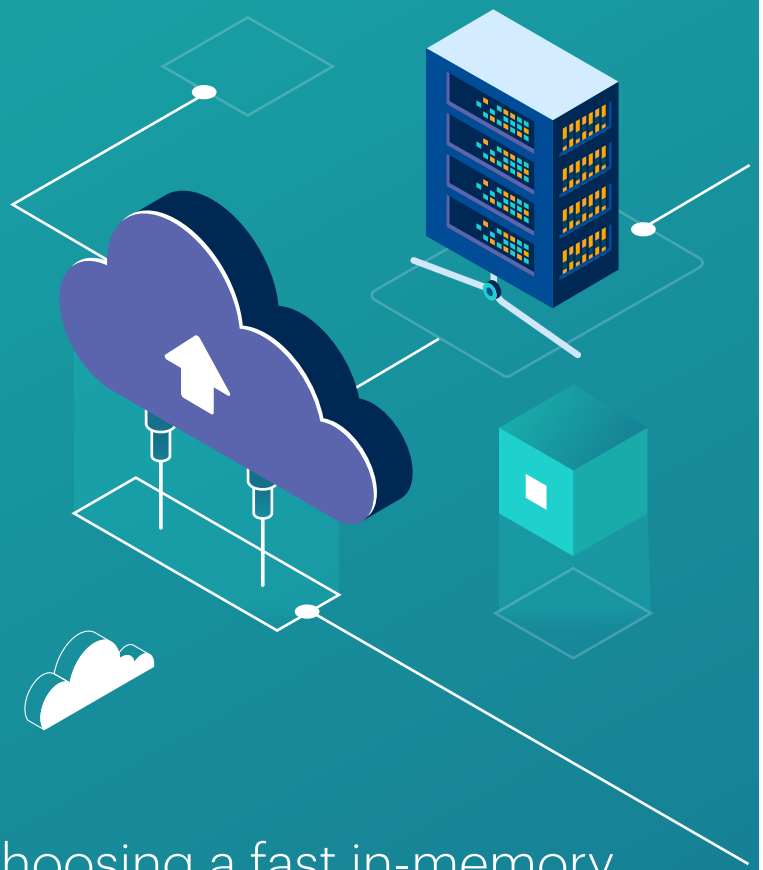
Build upon an in-memory database.

While you can get a solid speed boost from an in-memory cache, nothing compares to the sub-millisecond latency of a fully in-memory database. Additionally, choosing an in-memory database that's easy to deploy and maintain on any enterprise cloud ensures another important speed advantage faster time to market.

Avoid proliferation of multiple specialized databases to manage complexity.

Developers have embraced the NoSQL paradigm for documents, graph, and other data models, and several non-relational databases are available to solve point problems in different domains. However, many of them run on disk or have other performance limitations that may prevent you from being fast enough for large-scale, real-time financial services applications.

> "Future generations of data tools, data models, open-source offerings, and managed services may offer better performance and a faster path to innovation than what any one cloud provider offers today."



> "Choosing a fast in-memory database that's easy to deploy and maintain on any enterprise cloud brings another important speed advantage—faster time to market."

Also, the more different databases you put into production, the fewer people in your organization truly understand them all—and even the best developers try to avoid frequent context-switching to avoid curtailing their productivity. Overall, having multiple databases entails more training costs, bigger teams, slower development, and more difficulty troubleshooting problems.

Evolve towards multiple data models.

The fastest growing, most important applications are now being built with multiple data models. That's why the ideal approach is to build on a single in-memory database that can handle multiple data models, with a unified operational interface. This represents a highly effective evolution in databases, well adapted for where the marketplace is headed.

Multi-cloud and hybrid-cloud capabilities are essential.

Cloud providers such as Amazon, Microsoft, and Google have integrated their preferred technology approaches into their cloud offerings. Although emphasizing a best-of-breed approach, they tend to steer you toward their respective managed database services, which may or may not be your ideal solution. Future generations of data tools, data models, open-source offerings, and managed services may offer better performance and a faster path to innovation than what a particular cloud provider may be committed to offering today. That's why many companies are working to preserve their options with multi-cloud environments.

Get started now!

Fintech startups and their technology partners are already building services using modern technology stacks, including real-time databases with multiple data models. Established financial institutions are acknowledging the trend with technology roadmaps that anticipate the need for real-time business requirements. It's not too early to begin with proof-of-concept experiments, test cases, and deployments of new workloads using in-memory databases. And if you don't get started soon, it could be too late to catch up.

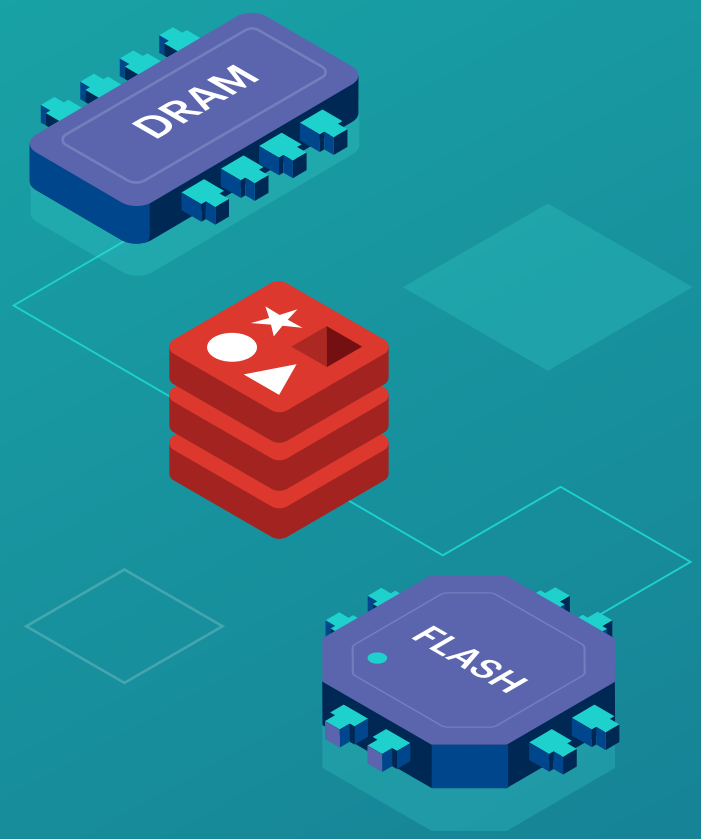
The role of Redis Enterprise in building real-time financial services

When it comes to building real-time financial services, it's all about performance. Unless the data layer is extremely fast, your applications simply won't be able to deliver the real-time performance your customers demand.

Redis is the only solution that delivers this kind of performance as an in-memory database with support for multiple date-models with dedicated engines. The most launched, most used, and most loved database in the world, Redis is beloved by software developers for its simplicity, flexibility, and extensibility. Unmatched performance and ease of development are major reasons why major financial institutions like Deutsche Börse, Xignite, and Vetr are already using Redis Enterprise to power real-time applications.

> "When it comes to building real-time financial services, it's all about performance."

redis



> **Redis on Flash** lets you utilize a tiered memory and storage approach to support large datasets with high database performance, at significantly lower cost.

Efficient performance at scale

Efficiently scaling database performance is critical for real-time financial services applications. Redis Enterprise scales linearly with infrastructure capacity, resulting in more resource-efficient databases without compromising throughput and latency. Furthermore, Redis Enterprise clusters can be scaled up without adding additional overhead, imposing downtime, or requiring changes to application code.

When you need real-time performance across larger datasets, Redis on Flash lets you utilize a tiered memory and storage approach to support large datasets with high database performance, at significantly lower cost than putting everything in DRAM.

Resilience, persistence, and high availability in any scenario

It's not enough to consistently deliver real-time results when everything is working perfectly. In order to ensure that those results are available at all times, even when there's an outage or other service interruption, financial institutions also need highly resilient databases with the means to instantly recover from failure without data loss.

Redis Enterprise provides several built-in high availability, durability, and disaster-recovery capabilities:

- Database replication options that protect against failures across multiple infrastructure regions or datacenters.
- Automated failure detection and failover to protect against unplanned downtime, outages, and data loss.
- Policy-driven data persistence, backups to cloud storage, and automated cluster recovery.

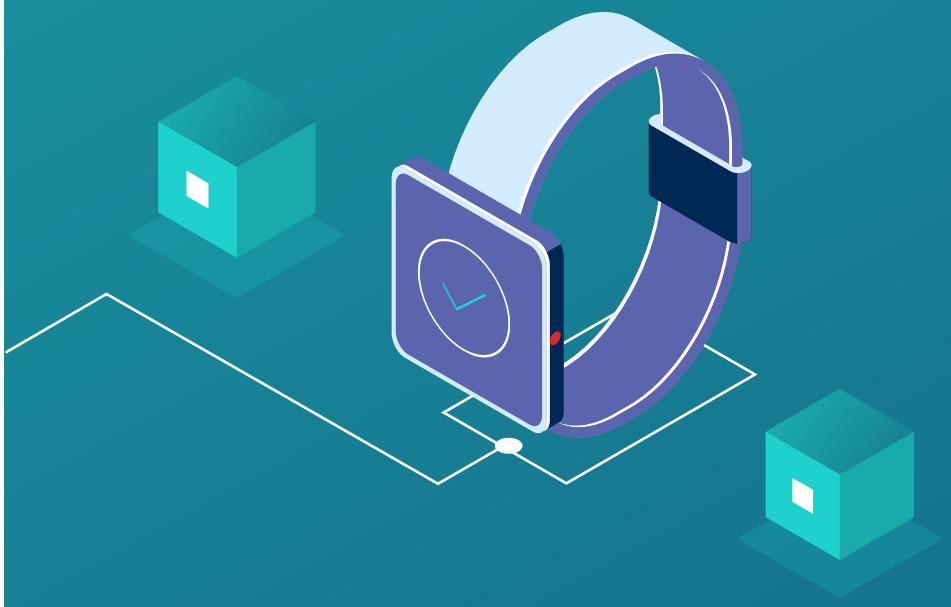
Multiple data models to power real-time use cases

Whether you're managing real-time payments, coordinating trading workflows, or detecting and mitigating fraud, real-time financial services use cases must be able to instantly process large volumes of data, using data models appropriate for the task at hand. As a database with a variety of native data structures and modern data models, Redis Enterprise is uniquely positioned for this challenge. Redis data structures let developers implement sophisticated functionality at the database level with fewer lines of application code, while Redis modules provide a variety of data models that can be readily applied to use cases such as full-text search, time-series data processing, social graphing, and many more.

Multi-cloud and geo-distribution for maximum flexibility

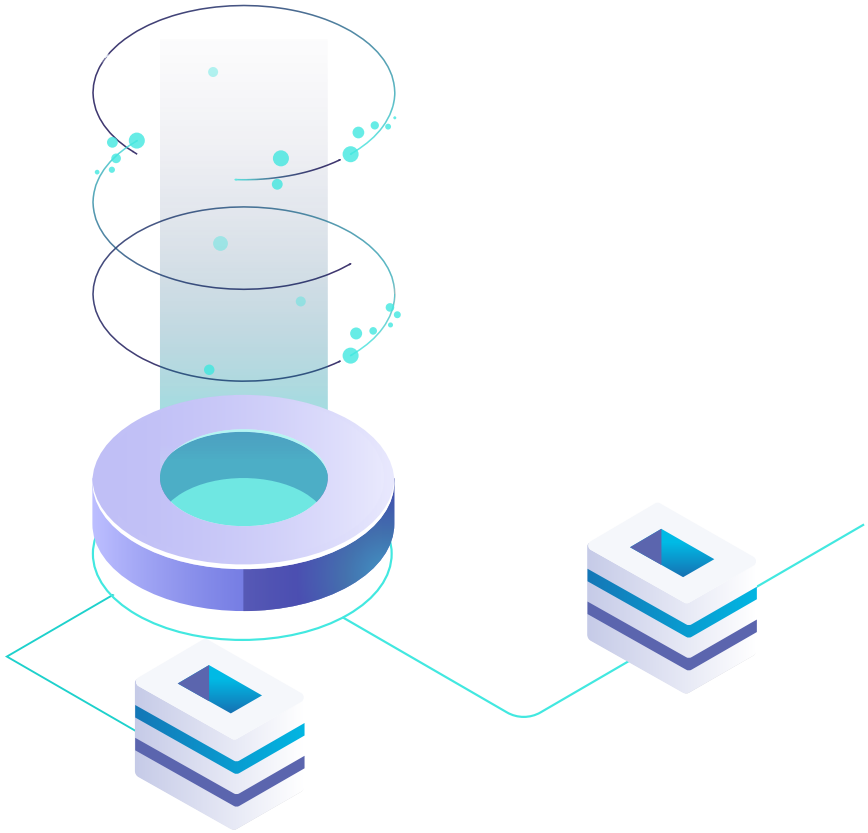
As the financial services technology landscape evolves, the data layer must have the flexibility to work with the technologies that support real-time capabilities, including a wide variety of multi-cloud and hybrid cloud environments. Redis Enterprise is available on Amazon Web Services, Google Cloud, and Microsoft Azure as a managed service, provides automation and support for common operational tasks, and integrates with the platforms underpinning modern software architectures, such as containers and Kubernetes.

> "It's not enough to consistently deliver real-time results when everything is working perfectly. Financial institutions also need highly resilient databases with the means to instantly recover from failure without data loss."



> Financial services firms need the flexibility to deploy their applications in a wide variety of multi-cloud and **hybrid cloud environments.**

Financial institutions running globally distributed applications also benefit from the ability to deploy Redis Enterprise as an Active-Active, geo-distributed database. The use of conflict-free replicated data types (CRDTs) enables applications to gracefully handle simultaneous updates from multiple geographic locations, powering use cases like fraud detection, rate limiting, and personalization on a global scale without compromising latency or availability for users.



The time to act is now

To get started, try building and testing a proof-of-concept real-time application. Deploy an in-memory database for new use cases and workloads. Powerful trends towards instant response are reshaping financial services, so getting your feet wet now is essential to make sure you're familiar with the key technologies needed to meet these challenges.

Here are some resources to help you start addressing some of the most pressing challenges in financial services applications:

→ Redis blog: What to Choose for Your Synchronous and Asynchronous Communication Needs

An overview of communication patterns commonly used in distributed systems, and what options to choose when connecting services.

→ Redis blog: Redis as a Primary Database

Expanding the role of Redis to a primary database can improve application performance and reduce operational complexity.

→ The New Stack: How to Support Large-Scale Analytics with Probabilistic Data Structures in Redis

Probabilistic data structures can improve the efficiency of large scale systems that need to quickly filter through data, such as for fraud detection.

→ Video: Solving Coordination Problems in Modern Service-Oriented Architectures

Developing modern, service-oriented architectures requires stateless services to enable horizontal scalability. This video explains how Redis Pub/Sub can be used to manage coordination between .NET services.

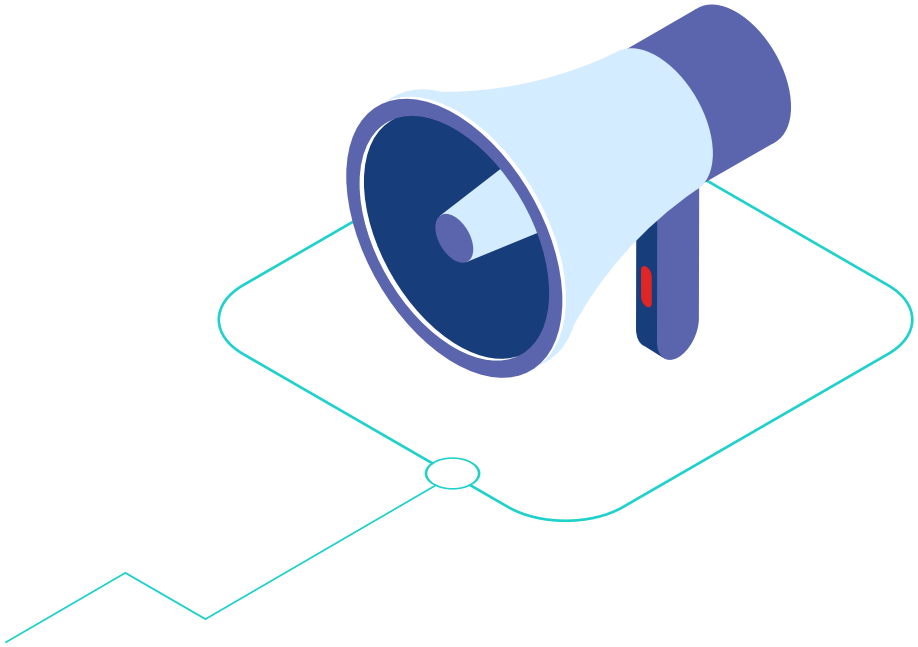
→ Video: Declarative Caching with Redis

While caching is widely understood as a concept, not all queries should be cached. Declarative caching with Redis can provide finer control over which results are cached and for how long.

→ Video: How to Scale Real-Time Applications Using Redis Pub/Sub

Learn more about the most common architecture for applications that push updates to their users in real-time using WebSockets, and how to scale them horizontally with Redis Pub/Sub.

To learn more about how financial services firms are leveraging Redis Enterprise to build real-time FinServe applications, visit Redis Enterprise for Financial Services, or check out how major financial institutions like Deutsche Börse, Xignite, and Vetr are already using Redis Enterprise. To get started, try Redis Enterprise in the cloud or download the Redis Enterprise Software for a free trial now.



Thank You

To learn more about how financial services firms are leveraging Redis Enterprise to build real-time FinServe applications, visit

Redis Enterprise

or check out how major financial institutions like Deutsche Börse, Xignite, and Vetr are already using Redis Enterprise.

To get started

Try Redis Enterprise in the cloud or download the Redis Enterprise Software for a free trial now.

Try The Free Trial

Follow Us:

redis.com