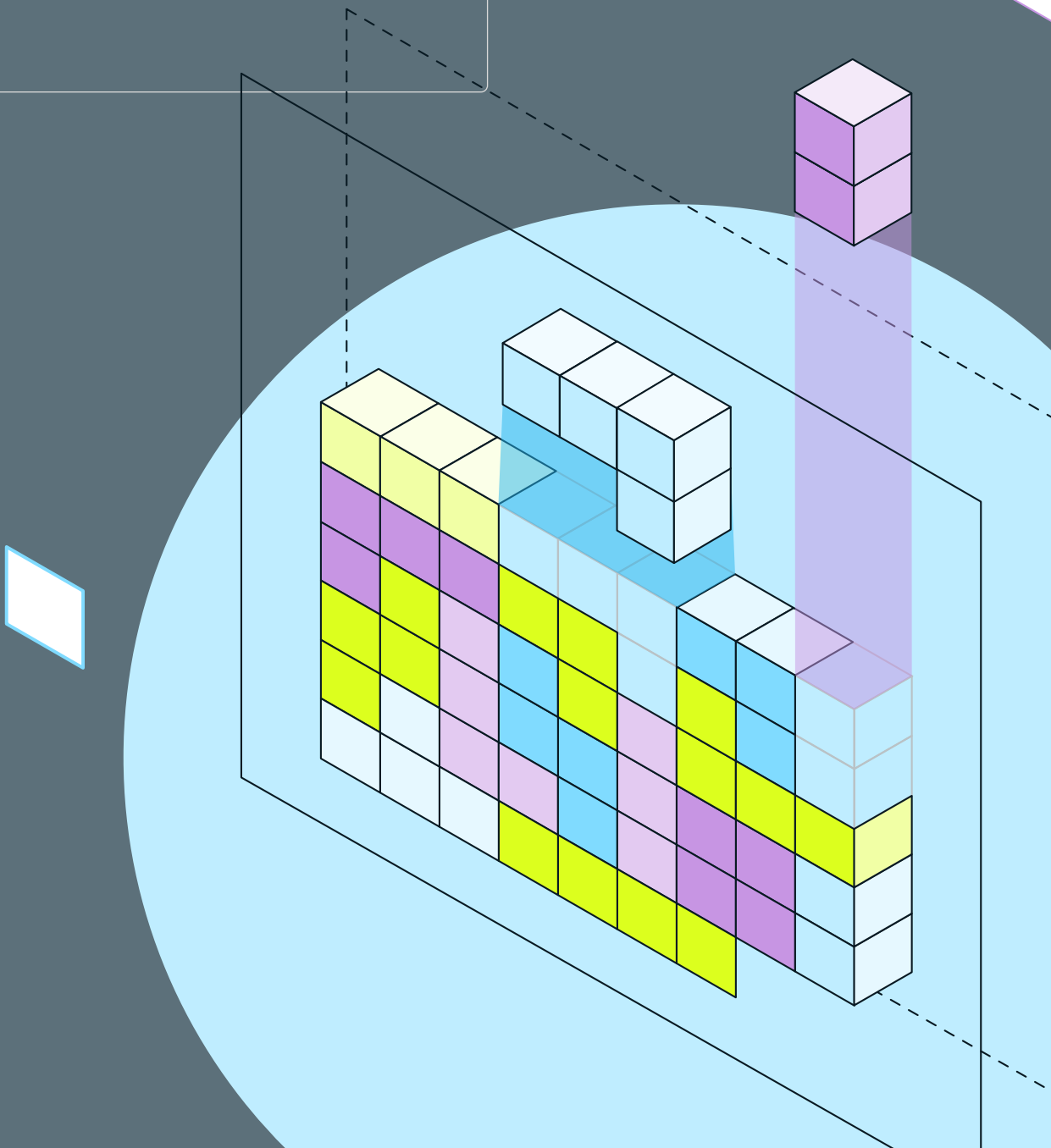
Redis

DATASHEET

# Your team isn't slow. Your data is just fragmented.

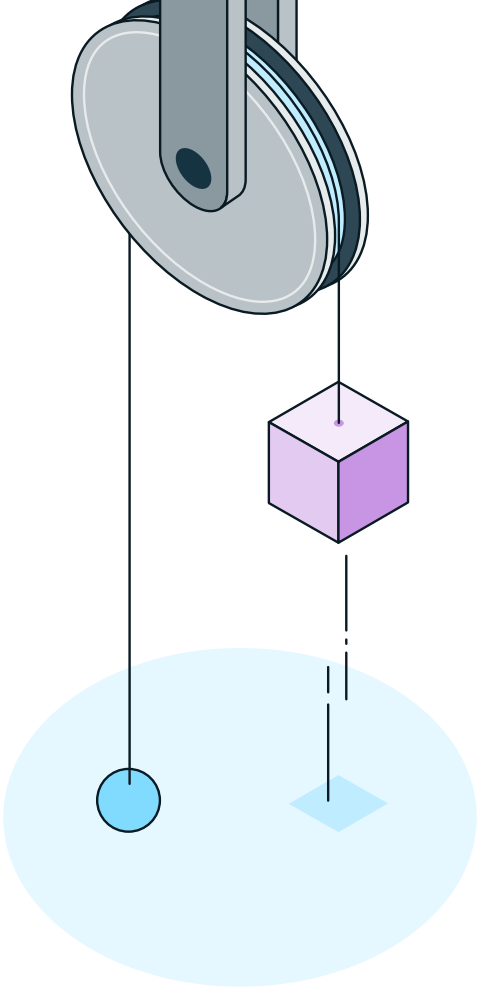


© 2026 Redis

Every new feature—from personalization and fraud prevention to AI search—adds another cache, vector store, or pipeline and more lines of glue code.

Teams end up maintaining infrastructure instead of innovating and shipping products.

It's not a tooling problem. It's a data architecture problem.



## The real blocker: A fragmented data stack

Modern apps need fast, fresh data for search, personalization, fraud prevention, and AI. But that data is spread across too many systems.

Every new feature adds pipelines, indexes, and failure points. Data falls out of sync. Responses slow down.

Engineering focus shifts from product innovation to infrastructure maintenance, reducing your ability to ship revenue-driving features.

## The difference is in the data platform

A unified data layer like Redis creates a single source for the memory every app needs. It streamlines product development and business growth in three key ways.

### Faster innovation

A single data layer for caching, search, vectors, and sessions means new features reuse the same patterns. Teams ship product instead of data plumbing.

### Business growth

Personalization only works when context stays current. Faster data means higher accuracy, better UX, and higher conversions to drive revenue.

### Confident scaling

Replacing a fragmented, over-provisioned stack with a single platform delivers low latency at peak loads, cutting infra cost as you grow.

> Redis is reliable, resilient, and built for scale. It powers our most critical systems—from pricing to personalization—while letting us focus on delivering innovation to our customers.

**Vivek Pariha,**
Head of Engineering, Purplle

## See the benefits side by side

| Without Redis | With Redis |
| --- | --- |
| → **Releases slow down**<br>Every feature needs new infra. Your team spends cycles on plumbing, not product. | → **Ship features, not infra**<br>One platform for cache, sessions, search, and vectors. New features reuse the same patterns, so no custom wiring or new services. |
| → **Stale data**<br>Multiple systems fall out of sync. AI features return inaccurate, inconsistent results. | → **Always-fresh context**<br>Redis automatically stays in sync with your databases meaning AI workloads get consistent, accurate data without manual pipelines. |
| → **Costs spiral at scale**<br>Each additional service compounds cost, complexity, and operational risk. | → **Predictable performance at scale**<br>Delivers consistent low latency under peak load, built-in high availability, and a single platform to operate instead of multiple. |

### The real-time context engine: Taking AI to prod faster

Most AI prototypes stall because each project needs its own vector store, cache, feature store, and custom wiring. Teams spend more time on infrastructure than building and delivering products.

As the real-time context engine, Redis delivers a low-latency, unified platform to search across every type of memory—from RAG and short/long-term memory to structured and unstructured data—to deliver the right answers to the right place at the right time.

We let your team build, get to production, and scale faster.