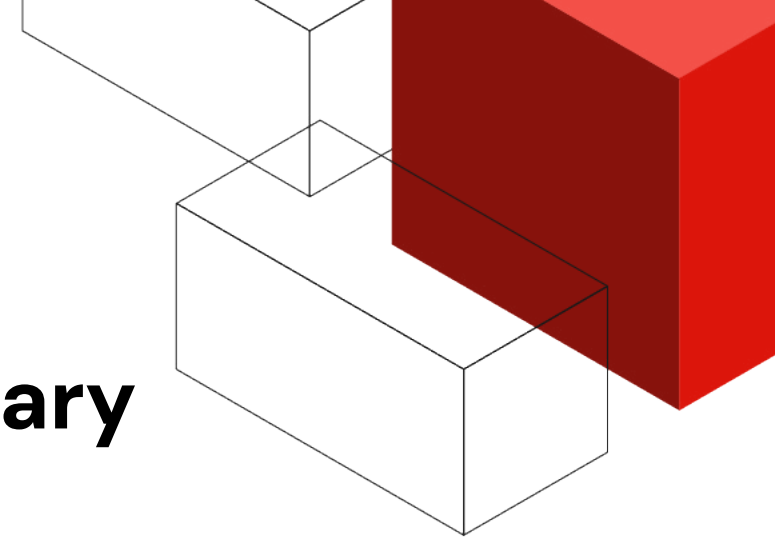
TiDB

WHITE PAPER

Scaling Multi-Tenant SaaS Platforms

Supercharge Growth with a Future-Proof Distributed SQL Database



Executive Summary

As SaaS companies accelerate growth, they face mounting challenges: Surging data volumes, performance constraints under high concurrency, multi-tenant complexity, and the relentless need for real-time insights. At the same, they need to manage infrastructure costs and maintain enterprise-grade reliability.

Traditional databases and legacy architectures often fall short, requiring manual sharding, redundant analytics pipelines, and costly infrastructure sprawl. These limitations hinder innovation, degrade customer experience, and threaten long-term scalability.

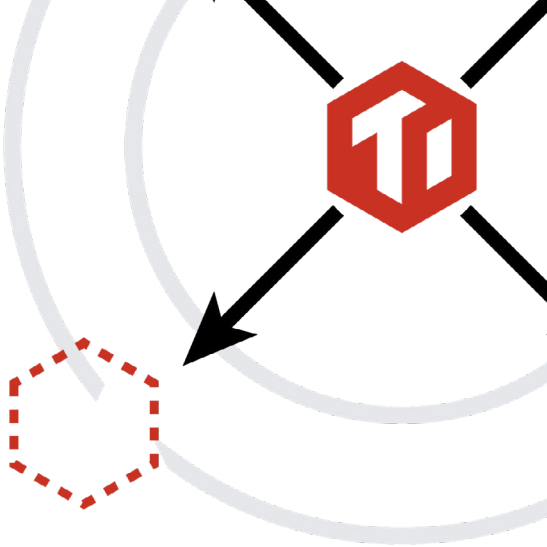
TiDB, a distributed SQL database built for multi-tenant SaaS scale, offers a fundamentally different approach. Designed to handle transactional and analytical workloads on a unified, cloud-native architecture, TiDB enables teams to:

- **Scale elastically** across compute and storage without rearchitecting.
- **Run real-time, multi-tenant applications** with logical isolation and fine-grained resource control.
- **Deliver sub-second performance** even with millions of tables and thousands of schemas.
- **Support always-on reliability** with 99.99% availability and online schema changes.
- **Consolidate OLTP and OLAP** to eliminate ETL complexity and unlock live operational insights.

Through real-world success stories from companies like Atlassian, Manus, and Conga, this paper illustrates how TiDB empowers SaaS leaders to simplify infrastructure, accelerate product velocity, and build future-proof platforms that scale with their ambitions.

If you're building or modernizing a SaaS platform, TiDB is the proven foundation to help you grow faster, operate smarter, and compete stronger in a data-intensive world.

Contents



Introduction

The SaaS model has revolutionized how software is delivered, enabling companies to scale faster, reach global markets, and deliver continuous innovation. But with this growth comes immense technical pressure: Managing massive data volumes across thousands of isolated environments, ensuring ultra-low latency for premium tenants, and maintaining always-on reliability while keeping infrastructure costs under control.

Traditional relational databases and NoSQL systems often fall short under these demands. Legacy architectures struggle to scale elastically, enforce strong consistency, and handle both transactional and analytical workloads in real time. For high-growth SaaS platforms, these limitations quickly become barriers to customer satisfaction, velocity, and profitability.

Enter TiDB, an open-source, distributed SQL database designed from the ground up to meet the unique demands of today's SaaS platforms. Combining the best of traditional RDBMS capabilities with the scalability of NoSQL and the power of mixed workload processing, TiDB enables SaaS platform companies to grow faster without rewriting their tech stack.

This white paper explores how TiDB elevates SaaS platforms to:

- Seamlessly scale across customers and workloads with a distributed architecture.
- Achieve high performance and availability, even under extreme concurrency.
- Optimize infrastructure costs with efficient multi-tenant data management.
- Deliver real-time operational insights without the overhead of separate OLAP systems.
- Maintain enterprise-grade security, compliance, and observability.

Through deep technical insights, real-world case studies, and practical best practices, this white paper provides a roadmap for SaaS technical leaders and architects to supercharge growth—backed by the proven capabilities of TiDB.

Whether you're building the next breakout SaaS platform or modernizing an existing platform, TiDB offers the foundation to scale confidently into the future.

CHAPTER 1

The Five Growth Challenges Facing Every SaaS Platform

The SaaS industry is expanding at an extraordinary pace. Leading platforms now serve tens of thousands of customers while managing petabytes of data across global infrastructure. But this rapid ascent is paired with mounting technical and operational pressures.

As SaaS platforms scale, they encounter a series of compounding challenges that can limit growth, hinder innovation, and strain profitability. Understanding these hurdles is essential for engineering teams, architects, and product leaders building the next generation of resilient, scalable platforms.

1. The Multi-Tenant Complexity Wall

Multi-tenancy is essential for SaaS platform scale but inherently difficult to manage. Traditional databases force you to choose between managing thousands of separate databases (silos) or one giant, messy database (shared). Each tenant must have confidence that their data is secure, isolated, and performing reliably. However, shared infrastructure often leads to contention for resources, unpredictable performance, and security concerns, especially in the presence of "noisy neighbor" workloads. Without granular controls over resource allocation, access, and visibility, SaaS platforms face challenges in ensuring fairness, enforcing SLAs, and maintaining compliance across customer environments.

2. Scaling Limitations

Most SaaS platforms now support over 20,000 customers—some well beyond 36,000. Each additional customer contributes more data, more concurrent users, and more complex usage patterns. Traditional architectures, especially monolithic or vertically scaled databases, struggle to keep pace. Scaling often requires disruptive downtime, manual intervention, or significant infrastructure overhauls. As a result, teams face bottlenecks that limit their ability to onboard customers quickly, respond to product demand, or launch into new markets.

3. Performance Bottlenecks

With SaaS platforms, performance is everything. Users expect fast, responsive experiences regardless of how many others are on the platform. But high concurrency and growing data volumes often lead to slow query performance, increased latency, and uneven load distribution. As workloads intensify, isolated

performance issues can snowball—affecting critical workflows and increasing churn risk. Ensuring low-latency, high-throughput performance becomes increasingly complex as scale increases.

4. High Availability & Disaster Recovery

SaaS platform users expect uptime as a given, not a feature. Any disruption—whether from hardware failures, software bugs, or regional outages—can lead to lost revenue and damage customer trust. SaaS teams must implement robust disaster recovery, seamless upgrade paths, and proactive monitoring just to meet baseline expectations around service continuity.

5. Cost vs. Performance Trade-offs

Profitability remains under pressure for most SaaS providers, with variable infrastructure costs consuming up to 80% of revenue in many cases. This creates a delicate balancing act between ensuring platform performance and keeping gross margins healthy. Legacy database sprawl, isolated tenant environments, and duplicative infrastructure for analytics can drive up costs significantly. The need to deliver consistently high-quality service while managing cost growth is a persistent source of friction for engineering and finance teams alike.

CHAPTER 2

Why TiDB is the Future-Proof Standard for Multi-Tenant SaaS Platforms

As SaaS platform providers scale to support thousands—or even millions—of end users, they face infrastructure limitations that legacy databases weren't built to handle. TiDB was purpose-built to support the demanding requirements of high-growth SaaS businesses. Its architectural innovations directly solve the pain points discussed in the previous chapter, enabling companies to grow faster, operate more efficiently, and deliver better user experiences.

Multi-Tenancy: Efficient, Isolated, and Secure

Multi-tenancy is foundational for SaaS platforms, allowing providers to serve many customers from a single platform while keeping their data and workloads isolated. TiDB excels here by supporting millions of tables and schemas in a single cluster, giving each tenant its own logically separated environment—without the overhead of maintaining separate database instances. TiDB's native multi-tenancy ensures your architecture remains clean and manageable, whether you have 10 tenants or 10,000.

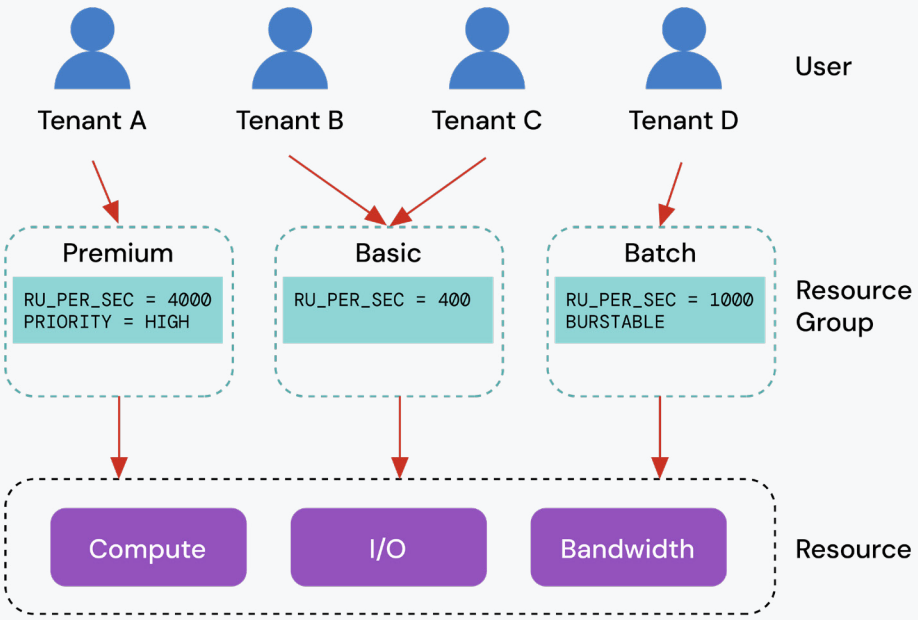


Figure 1. A multi-tenant architecture containing TiDB resource control features to achieve resource isolation and prevent noisy neighbors.

TiDB also offers fine-grained resource control. Resource groups allow operators to assign quotas and priorities to specific tenants, preventing "noisy neighbor" issues and ensuring consistent performance even under heavy load. With role-based access control (RBAC) and audit logging, security and compliance are built in.

Scaling to Millions of Tenant Tables: A New Standard in Scalability

Conventional relational databases often impose hard limits on table counts or experience performance degradation as schema counts grow. TiDB rewrites these rules. Thanks to its innovative distributed row storage engine and metadata manager, it can scale out to millions of tables across thousands of tenants while solving the limitations of "schema-per-tenant" designs.

Case in point: Atlassian tested TiDB 8.2 with 2.3 million tables and 4,000 schemas, experiencing zero performance degradation. For SaaS platforms where each customer might require their own isolated schema, this is a game-changer.

99.99% Availability: Built for Always-On SaaS Platforms

SaaS platforms are always-on, and any downtime translates to lost revenue and eroded trust. TiDB

provides four nines of availability (99.99%) through its self-healing, distributed architecture. Clusters can span multiple availability zones, automatically detecting and recovering from node or regional failures.

Additionally, features like online schema changes, rolling upgrades, and point-in-time recovery (PITR) enable continuous operations, even during maintenance and deployments. This makes TiDB ideal for mission-critical SaaS workloads.

Auto-Scaling: Infrastructure That Grows with You

SaaS platform growth is rarely linear. Customer acquisition campaigns, feature launches, or seasonal traffic spikes can rapidly change infrastructure needs. TiDB's architecture is designed to scale both compute and storage layers independently, enabling on-demand scaling without downtime or rearchitecting.

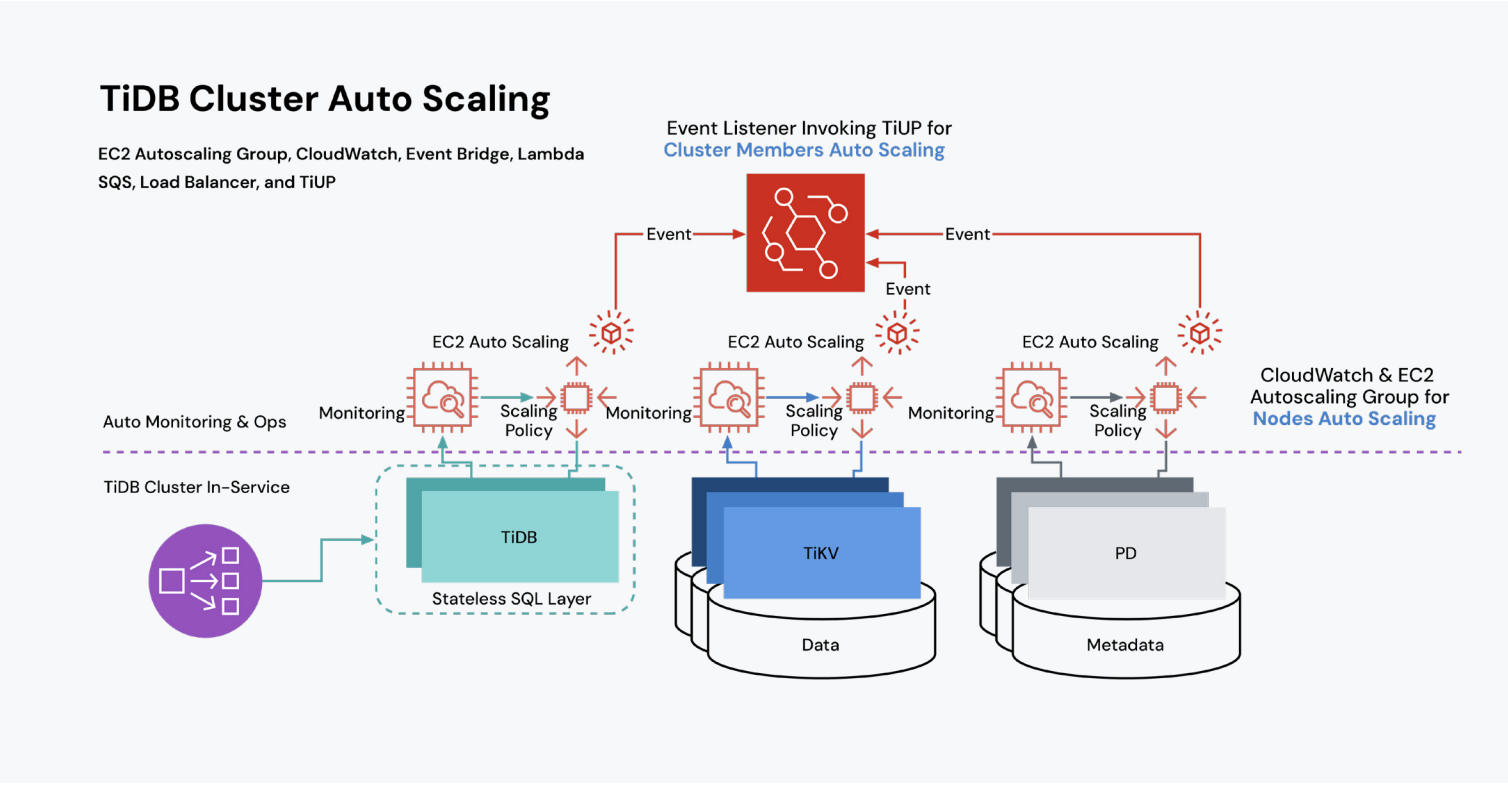


Figure 2. A diagram showing TiDB's auto-scaling architecture using various AWS services to handle tenant usage spikes.

TiDB automatically rebalances data across nodes to avoid hotspots and ensures optimal performance as new hardware is added. This elasticity supports both planned growth and sudden surges in usage—delivering the flexibility today's SaaS platforms demand.

Parallel Query Execution: Powering High-Concurrency, Low-Latency Experiences

SaaS platform users expect real-time responsiveness even when performing complex queries across massive datasets. TiDB meets these expectations with parallel query execution across distributed compute nodes. Every query is split and processed concurrently by multiple nodes, dramatically reducing response times for heavy workloads.

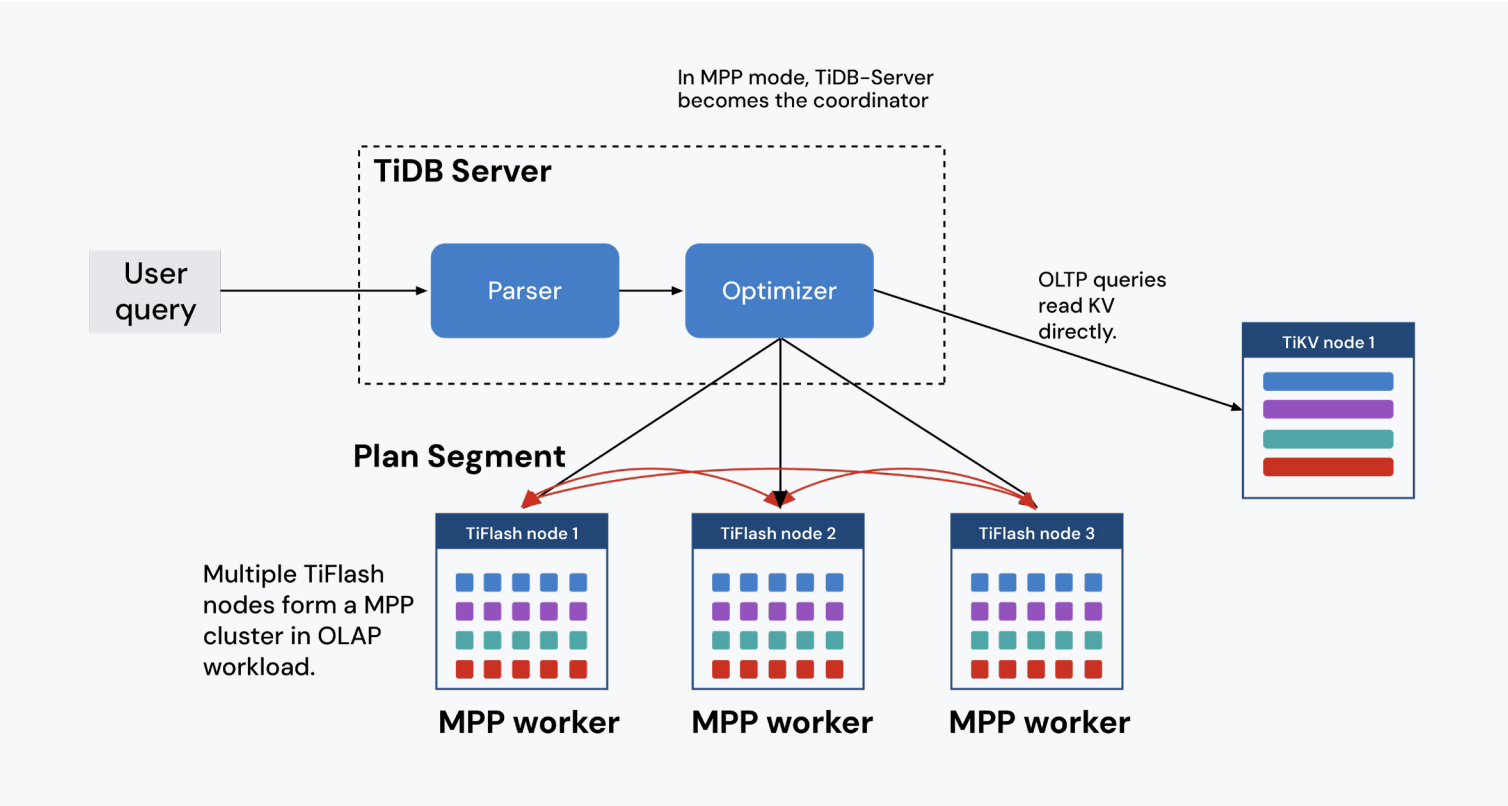
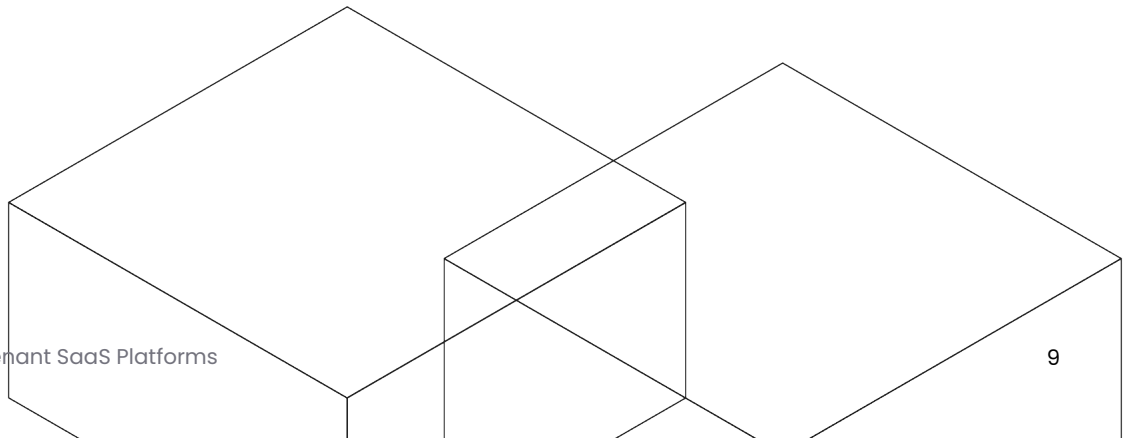


Figure 3. How Massively Parallel Processing (MPP) works in TiDB.

Whether it's generating real-time dashboards, running multi-tenant analytics, or serving a global user base with sub-second latency, TiDB delivers performance at scale without compromising on ACID guarantees or SQL compatibility.



CHAPTER 3

How TiDB Supercharges Multi-Tenant SaaS Growth

High-growth SaaS platforms live and die by their ability to scale—across users, data, geography, and features. Scaling isn't just about adding more hardware; it's about building a flexible, intelligent data infrastructure that evolves with your business. TiDB's distributed SQL architecture is purpose-built for this challenge, helping SaaS platforms scale with confidence, performance, and precision.

Elastic Scalability for Unpredictable Growth

SaaS platform growth is often unpredictable. New customer acquisitions, product launches, or viral adoption can cause sudden surges in demand. Traditional monolithic databases struggle to handle such dynamism due to rigid architecture and vertical scaling limits.

TiDB solves this with elastic scalability through auto-sharding and distributed architecture. Rather than relying on a single node, data is automatically partitioned and distributed across row-based storage nodes, and workloads are balanced across compute layers. Horizontal scaling of both compute and storage can happen independently and without downtime, ensuring that infrastructure never becomes a bottleneck.

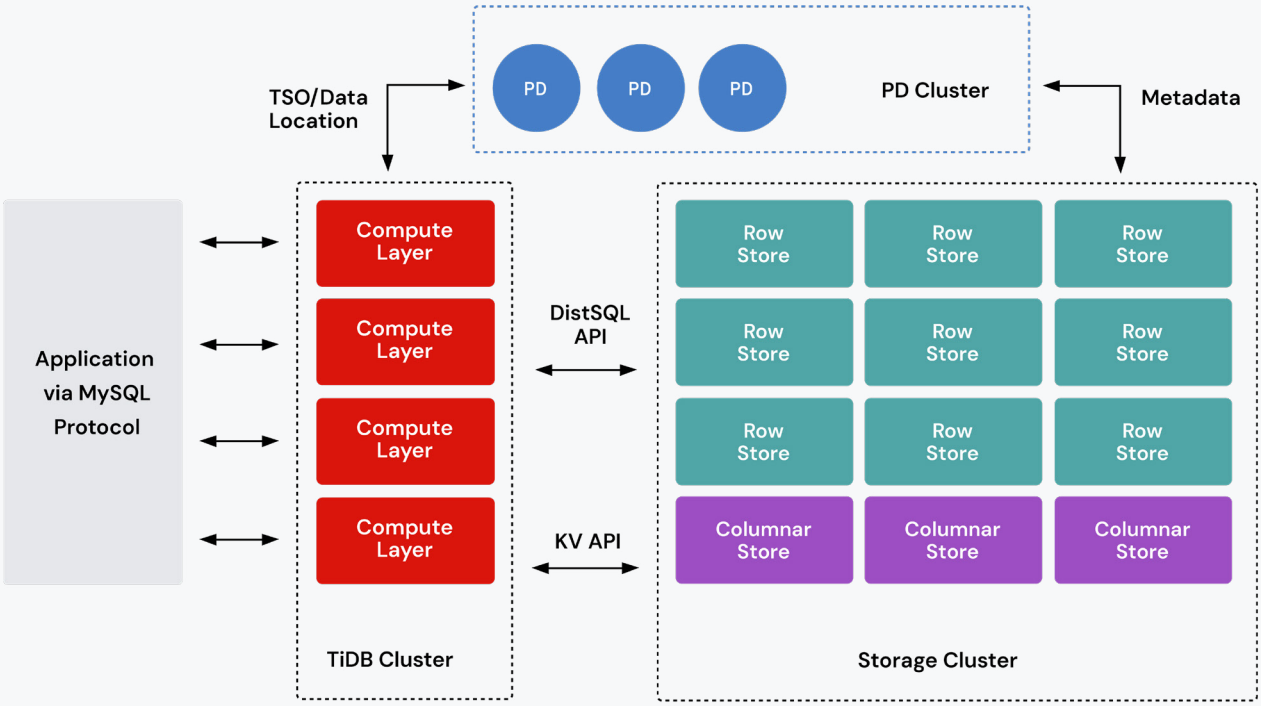


Figure 4. A diagram depicting TiDB's distributed SQL architecture that separates compute from storage.

Unlike monolithic RDBMSs that require complex rewrites or manual sharding at scale, TiDB grows seamlessly as your business grows—supporting petabyte-scale datasets and millions of concurrent users.

Enhancing Performance at Scale

As SaaS platforms grow in complexity and users, maintaining consistent performance becomes a challenge. Bottlenecks arise when too many operations are concentrated on a single database node.

TiDB avoids this trap by distributing data and workload across multiple nodes, enabling real load balancing for both transactional and analytical operations. Its separation of compute and storage layers ensures that performance scales linearly with hardware.

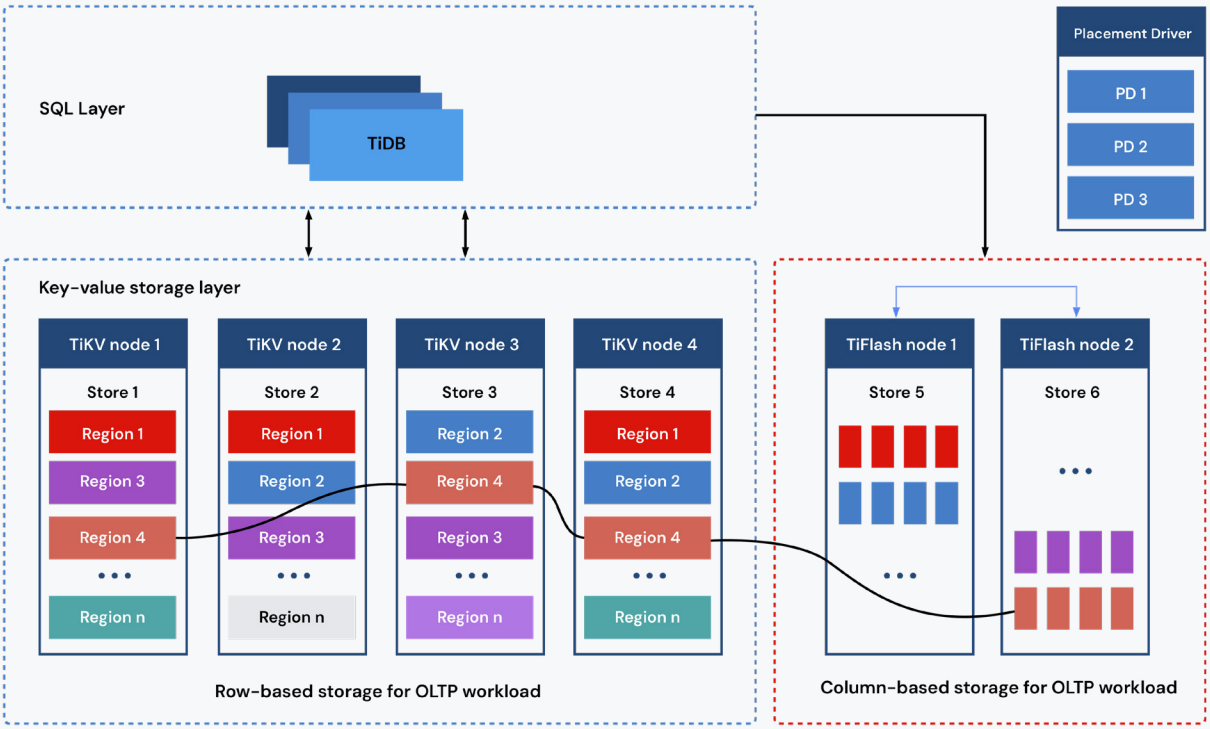


Figure 5. A diagram showing TiDB's storage layer consisting of row-based and column-based storage.

Additionally, TiDB integrates row-based storage for high-throughput OLTP workloads and column-based storage for real-time analytics. These components work together to optimize read and write paths, providing fast response times even under high concurrency.

This hybrid capability ensures that SaaS platforms don't have to choose between performance and flexibility. They get both.

Enabling Multi-Tenant Efficiency

Supporting thousands of customers in a shared environment is central to SaaS economics. But multi-tenancy comes with challenges: How do you isolate tenant data, maintain performance, and control resource usage?

TiDB supports millions of tables per cluster, allowing each tenant to have their own schema or set of tables. This approach eliminates “noisy neighbors”, making data management, backup, and security more granular and reliable. With TiDB, you stop spending engineering cycles fixing performance issues for premium tenants and start building features.

Additionally, TiDB offers resource control features that let operators define quotas for read/write throughput (RUs), ensuring fair usage and preventing noisy neighbor effects. This allows SaaS platform providers to optimize cloud costs while maintaining a high-quality experience across the board.

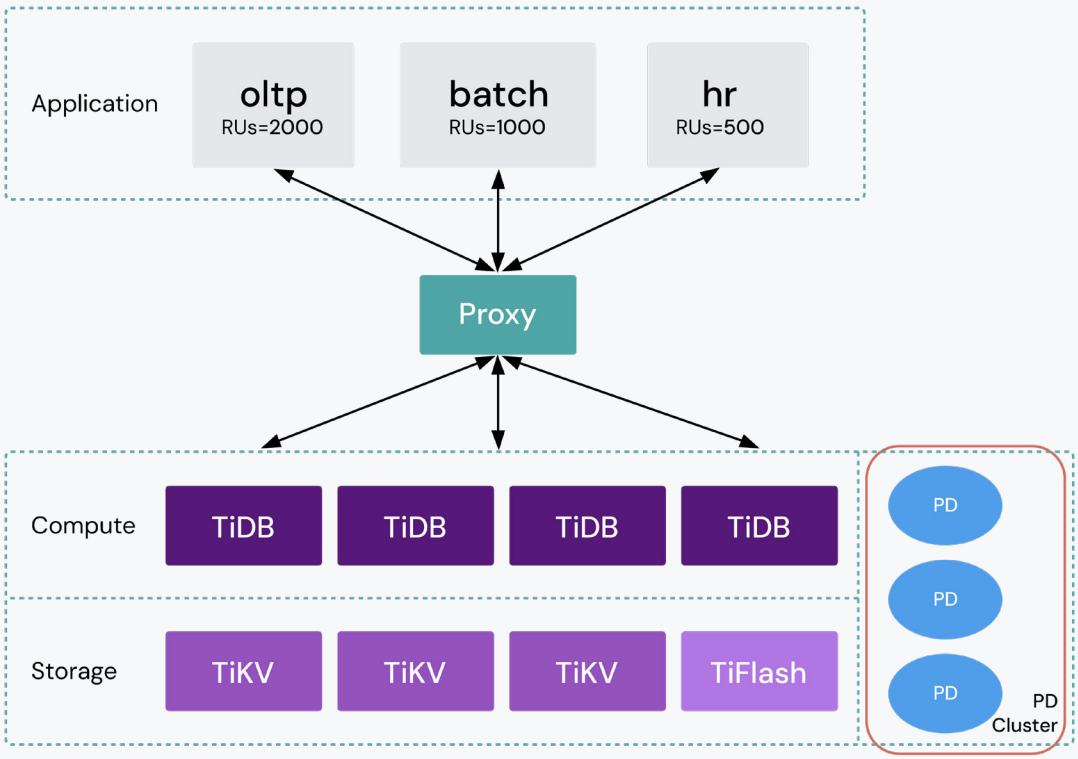


Figure 6. How TiDB allocates resources for each tenant to ensure consistent performance.

By combining logical isolation with physical efficiency, TiDB helps SaaS platforms build scalable, maintainable multi-tenant architectures.

Driving Real-Time Operational Insights

Today's SaaS users expect real-time insights, from in-app dashboards to usage analytics and billing metrics. However, most legacy databases are optimized for transactional workloads and require separate OLAP systems for analytics, resulting in ETL pipelines, data duplication, and delays.

TiDB eliminates this gap through its mixed workload processing capabilities. With column-based storage, transactional data is automatically replicated into a columnar format for real-time analytical queries, all with ACID compliance and zero ETL overhead.

This unified approach empowers SaaS platforms to offer instant analytics, drive intelligent product experiences, and improve operational decision-making without the complexity of multiple databases.

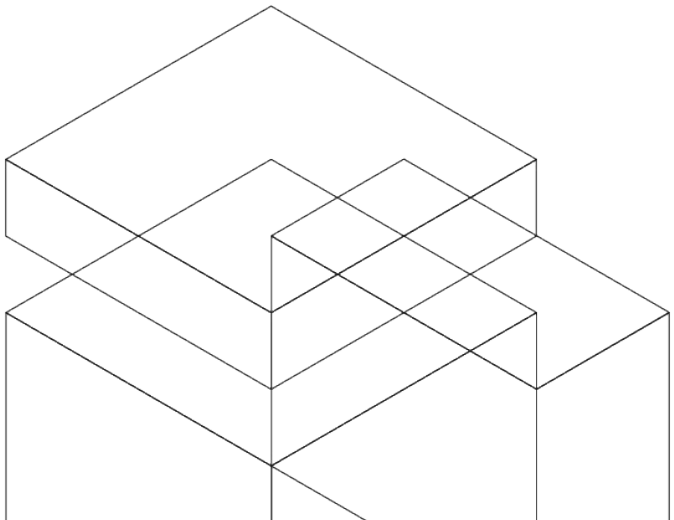
Ensuring Enterprise-Grade Reliability & Security

SaaS platform providers must meet enterprise demands for availability, reliability, and compliance. Down-time, data loss, or security breaches are unacceptable, especially for platforms handling sensitive business or user data.

TiDB is engineered for enterprise-grade reliability, featuring:

- **Multi-AZ deployments** for high availability (99.99% SLA).
- **Automated failover and data recovery** via region replicas and snapshots.
- **Rolling upgrades and online schema changes** to avoid downtime during updates.
- **End-to-end encryption**, both in transit and at rest.
- **Role-based access control (RBAC)** and **audit logging** for fine-grained security.
- **Point-in-time recovery (PITR)** and **log-based backups** for robust disaster recovery.

These capabilities ensure that SaaS platforms built on TiDB can meet the toughest SLAs and regulatory requirements, while maintaining operational agility.



CHAPTER 4

Real-World Success Stories

SaaS platform companies across industries turn to TiDB to solve real-world scaling, performance, and operational challenges. From reducing infrastructure complexity to dramatically boosting performance, TiDB arms innovators to build faster, leaner, and more resilient platforms. This chapter showcases how organizations like Atlassian, Manus, and Conga, transformed their SaaS platform architectures with TiDB at the core.

Atlassian

Scaling Millions of Tables for the Ultimate Multi-Tenant Platform

Atlassian's Forge platform empowers developers to build and run secure apps directly within Atlassian's cloud products. To ensure strict data isolation and security for its massive user base, Atlassian employs a "schema-per-tenant" architecture. However, as the platform grew, this design created an extreme metadata challenge: managing millions of tables within a single database cluster.

The Challenge

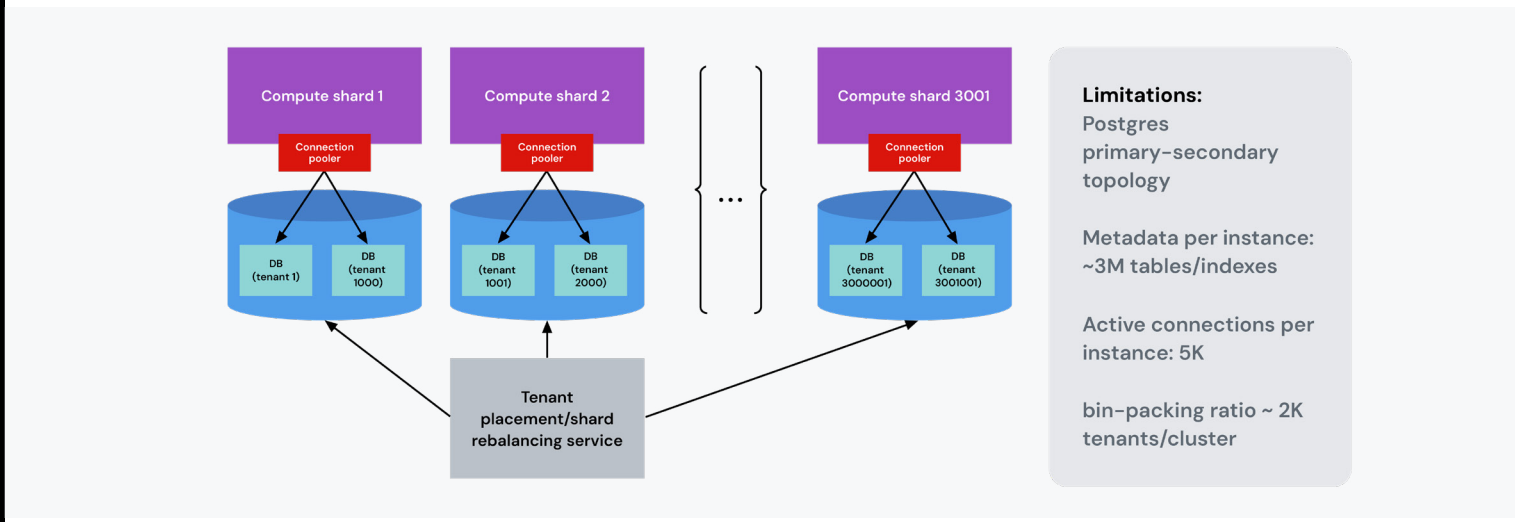


Figure 7. Atlassian's original data architecture featured hundreds of PostgreSQL clusters and complex shard rebalancing.

- **The "million table" wall:** Traditional databases struggled to handle the metadata overhead of a schema-per-tenant model, hitting performance limits at just a few thousand tables.
- **Operational paralysis:** Simple schema changes (DDL), such as adding a column to 100,000 tables, took over 6 hours, severely slowing down feature rollouts.
- **Infrastructure sprawl:** Scaling with legacy systems would have required deploying and managing thousands of separate database instances, creating a nightmare of complexity and cost.

The Solution

Atlassian chose TiDB to support its massive multi-tenant architecture after it proved capable of hosting over 3 million tables in a single cluster.

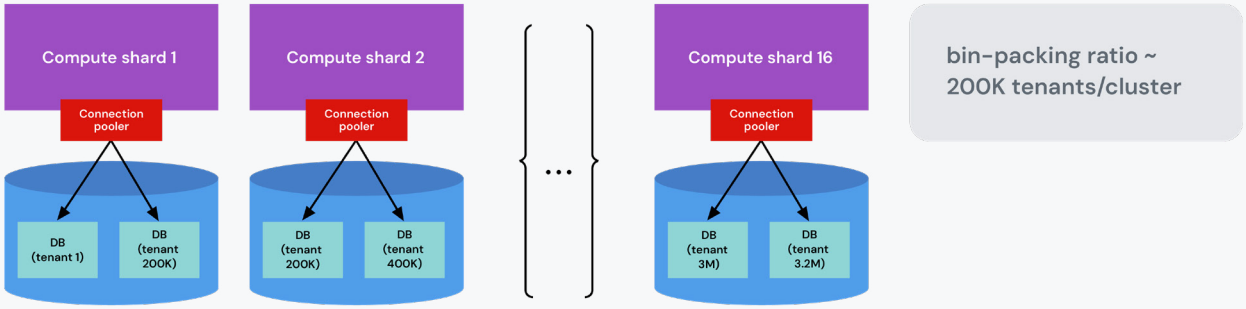


Figure 8. Atlassian's updated data architecture with TiDB at its core, leading to database consolidation from 750 Postgres clusters to only 16 TiDB clusters.

- **Extreme metadata scalability:** TiDB's architecture decouples compute from storage, allowing it to manage millions of table schemas without performance degradation.
- **Non-blocking schema changes:** TiDB's online DDL capabilities allowed Atlassian to execute widespread schema updates across thousands of tenants without downtime.
- **Unified management:**, By consolidating millions of tables into one cluster, Atlassian eliminated the need to manage thousands of individual database instances.

The Results

- **50x faster schema creation:** Reduced the time to create 1 million tables from 2 days to under 2 hours.
- **Accelerated iteration:** Cut the time required to add columns to 100,000 tables from 6 hours to just 32 minutes.
- **Operational efficiency:** Reduced cluster restart times by 90% (from 30 minutes to 3 minutes), ensuring higher availability for customers.
- **Simplified backups:** Slashed the time required to back up 1 million tables by 75%.

> *"With TiDB, we can now pack up to 100,000 tenants per instance, which resulted in a huge reduction of our Postgres fleet. So, instead of a thousand Postgres instances we just need 16 TiDB clusters to support our plugin platform." —* **Atlassian**

manus

Powering Agentic AI with Instant, Serverless Scalability

Manus is a general-purpose agentic AI platform that allows users to build full-stack apps from a single prompt. Because its AI agents "context switch" rapidly and iterate through thousands of steps to build software, Manus needed a database that could act as a dynamic "long-term memory" for every agent.

The Challenge

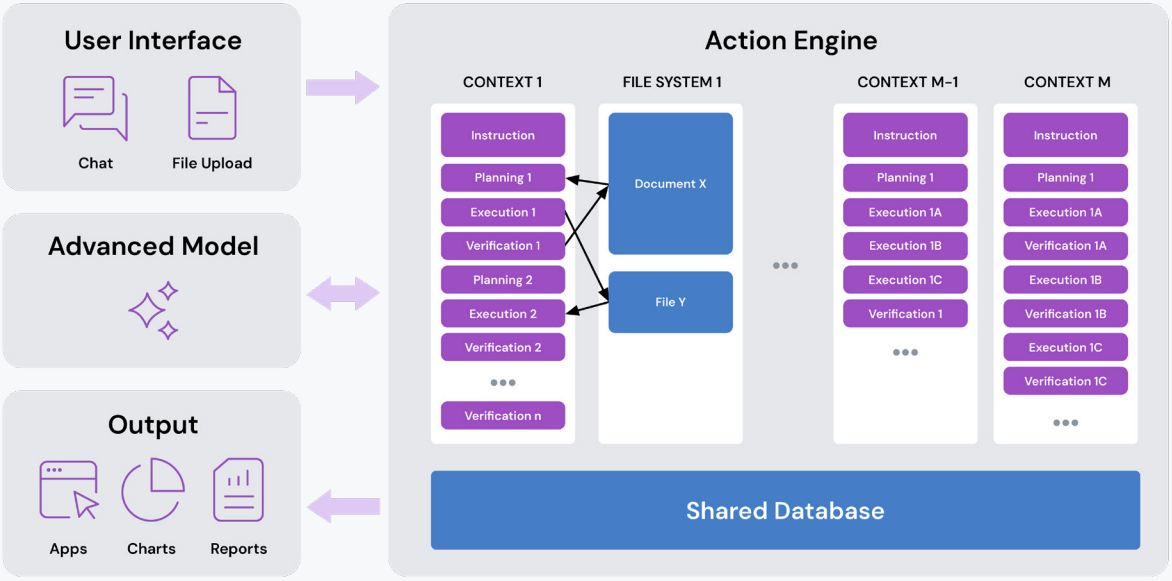
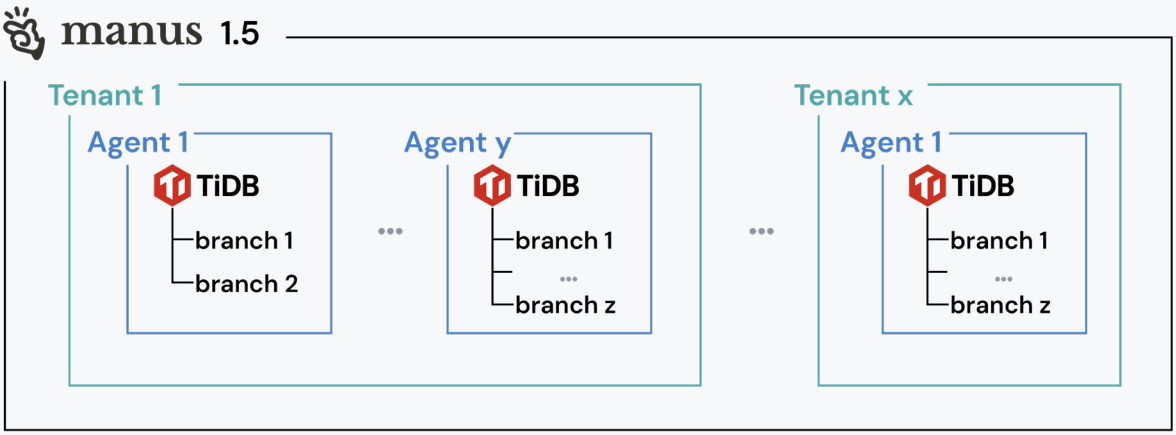


Figure 9. The Manus Action Engine's core cognitive loop. Each observation is persisted to the database (acting as Long-Term Memory), forming an ever-growing context that feeds back into the next decision cycle.

- **Viral growth:** Manus accumulated a waitlist of over two million users in weeks, creating unpredictable spikes in demand that monolithic databases couldn't handle.
- **Agent isolation:** Every AI agent requires its own isolated environment to branch, test, and iterate on code without risking data corruption or "noisy neighbor" interference.
- **Complex context engineering:** Agents generated thousands of stateful iterations per task, demanding high-throughput writing and low-latency state reconstruction.

The Solution

Manus migrated to TiDB to create a "Database-as-Code" architecture where AI agents can spin up and manage their own databases.



- **Instant branching:** Agents can instantly "fork" a database to test risky hypotheses in a sandboxed environment without affecting the main application.
- **Serverless elasticity:** TiDB's serverless architecture allows Manus to spin up tens of thousands of isolated database clusters in seconds, paying only for active processing.
- **MySQL compatibility:** TiDB enabled a rapid migration in just two weeks without major system redesigns, allowing the team to meet their public launch timeline.

The Results

- **Massive parallelism:** Enabled thousands of AI agents to work simultaneously in isolated environments without performance degradation..
- **Cost efficiency:** The serverless "Request Unit" (RU) model ensured Manus only paid for active compute, making it economically viable to run thousands of ephemeral experiments.
- **Zero-ops scale:** The team avoided managing shards or provisioning hardware, allowing them to focus entirely on improving their AI models.
- **Future-proof foundation:** Provided a limitless memory canvas that scales horizontally as agent complexity increases.

> *"Our goal was to find a database system that could scale seamlessly in both compute and storage without major system re-designs, to survive the present and finally unlock our full public launch. TiDB was that database." —* manus

conga

Reinventing Revenue Operations with Unified Data Infrastructure

Conga delivers an end-to-end suite of SaaS products for revenue lifecycle management—from contract lifecycle and CPQ to billing and document automation. Originally built on Salesforce and traditional RDBMS infrastructure, Conga faced challenges adapting to growing complexity, scalability demands, and multi-cloud environments.

The Challenge

- Scaling transactional and analytical workloads for a diverse global customer base, including highly regulated industries.
- Managing separate data stacks for OLTP, OLAP, search, and AI use cases—resulting in operational overhead and high costs.
- Ensuring compliance, performance, and availability across a cloud-agnostic, CRM-independent platform.

The Solution

Conga adopted TiDB to unify their data platform. TiDB was selected after a rigorous evaluation, driven by key criteria:

- **Horizontal scalability** and strong performance under mixed workloads.
- **Open-source flexibility** to support on-prem, hybrid, and multi-cloud environments.
- **Real-time analytics** and simplified architecture with mixed workload processing.
- **Support for advanced features** like vector search and full-text indexing.

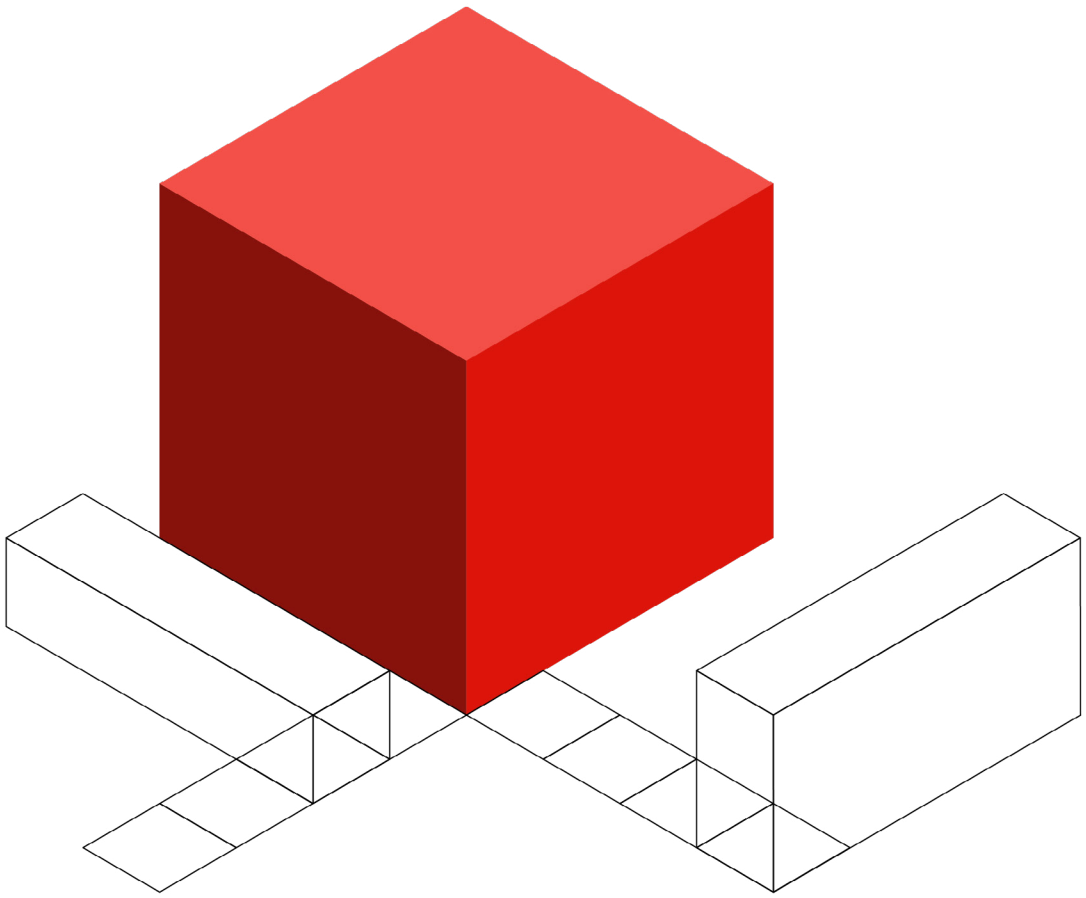
Conga implemented TiDB in phases, starting with analytical workloads for select customers. This cautious approach ensured stability, performance tuning, and feedback collection before expanding to transactional use cases.

The Results

- **Consolidated four different technologies** into one platform with TiDB.
- **Reduced infrastructure costs** and maintenance complexity.
- **Enabled real-time data insights** without ETL pipelines.
- Accelerated delivery of **AI-driven features** like personalized pricing and intelligent document recommendations.
- Improved **developer productivity** with MySQL-compatible interfaces.

> *Previously, we maintained four different technologies for handling transactions, analytics, search, and vector data. Now, TiDB offers a unified solution for all these use cases."*
>
> — conga®

These success stories underscore TiDB's ability to scale with the business, reduce infrastructure overhead, and enable real-time innovation. Whether you're a startup preparing for rapid growth or an established player modernizing legacy systems, TiDB provides the foundation to build the next generation of SaaS applications.



CHAPTER 5

Best Practices for Deploying Multi-Tenant Architectures

While TiDB offers powerful capabilities out of the box, achieving optimal performance, reliability, and cost efficiency in a SaaS context requires careful deployment planning and ongoing optimization. This chapter outlines proven best practices across architecture, performance tuning, and observability to ensure your TiDB deployment can meet the evolving demands of a high-growth SaaS environment.

Deployment Strategies: Cloud-Native vs. On-Prem Options

TiDB offers flexible deployment models tailored to the needs of SaaS providers:

Cloud-Native (Recommended for Most SaaS)

- TiDB Cloud (Dedicated or Starter) simplifies operations by abstracting cluster management, backup, and scaling.
- Ideal for fast-growing SaaS platform companies that want to focus on product development rather than infrastructure.
- Supports multi-cloud deployments so you can work with your infrastructure provider of choice.
- Built-in tools for auto-scaling, data encryption, and role-based access control.

Best for: Fast-moving SaaS teams prioritizing agility, elasticity, and reduced operational overhead.

Self-Managed (On-Prem or Hybrid Cloud)

- Offers full control over configuration, networking, and compliance.
- Useful for SaaS providers with strict data residency requirements, or those needing to run in private cloud environments.
- Requires in-house expertise for cluster maintenance, upgrades, and tuning.

Best for: Enterprises with advanced DevOps maturity or regulated environments.

Tip: Even in self-managed environments, adopting a containerized, Kubernetes-based deployment using TiDB Operator accelerates automation and simplifies scaling.

Performance Tuning Tips: Indexing, Partitioning, and Caching

Maximizing performance in TiDB requires thoughtful schema design and runtime tuning:

Indexing

- Use composite indexes for multi-column queries.
- Minimize over-indexing to reduce write amplification.
- Leverage covering indexes to eliminate unnecessary reads from the storage layer.

Partitioning

- Use range or hash partitioning to break large tables into manageable chunks and improve parallelism.
- Enables efficient pruning during query execution, especially important for time-series and multi-tenant workloads.
- Partitioned tables can be stored across different row store nodes for distributed execution.
- Use the Tenant ID as a partition key to improve query locality and simplify data pruning.

Caching

- Use TiDB's internal plan cache to reduce planning overhead for repeat queries.
- Implement application-level caching (e.g., Redis, Memcached) for frequently accessed metadata or computed values.
- Consider result set caching for dashboards and analytics that don't require real-time freshness.

Tip: Run frequent EXPLAIN ANALYZE on critical queries to monitor performance and fine-tune indexes.

Observability & Monitoring

Robust observability is critical in a multi-tenant SaaS setting where downtime or degraded performance affects many customers:

Monitoring

- Use Prometheus and Grafana for real-time metrics on query latency, resource usage, and replication health.
- Monitor row store write/read latency, region distribution, and columnar store query workloads.
- Track slow queries using TiDB's internal slow query log and performance schema views.

Alerting

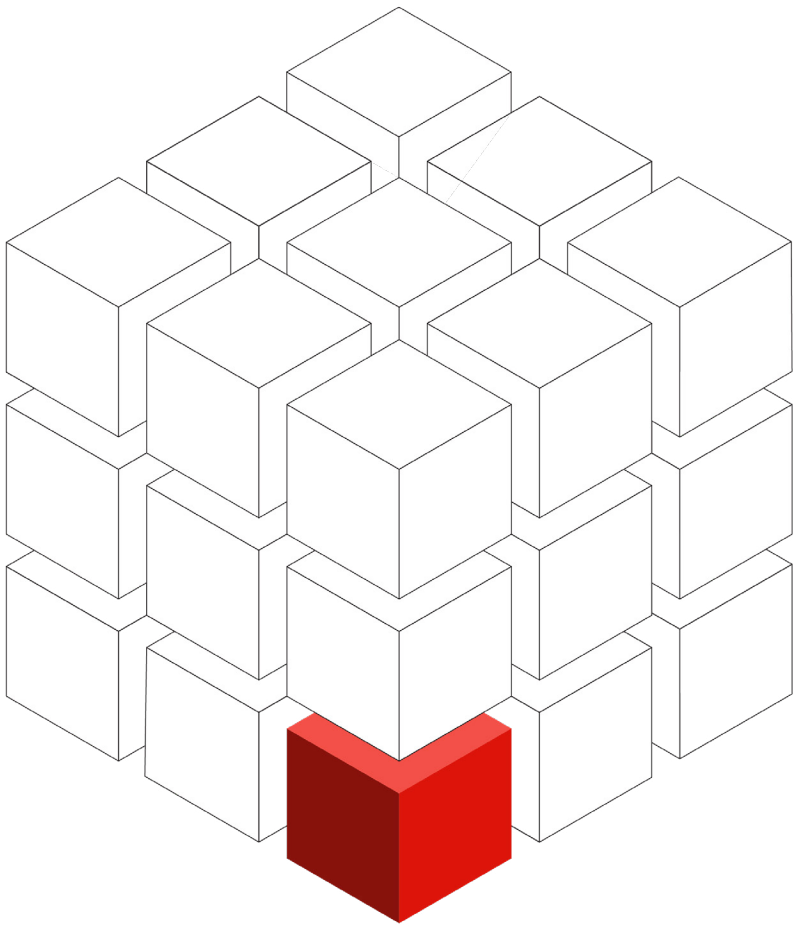
- Set up alerts for high query latency, region unavailability, storage hot spots, and replica lag.
- Ensure SLA violations are proactively detected before users are impacted.

Logging & Tracing

- Enable structured logging for TiDB compute, row store, metadata manager, and columnar store components.
- Use OpenTelemetry integration for distributed tracing of complex workflows.
- In TiDB Cloud, leverage built-in diagnostics and logging tools without additional setup.

Tip: Tag metrics with Tenant IDs to instantly identify which customer is driving load, rather than just seeing a generic spike in CPU.

By combining cloud-native deployment, strategic schema design, and robust observability, SaaS teams can get the most out of TiDB—ensuring their platform is scalable, performant, and reliable as customer expectations and data volumes grow.



Conclusion

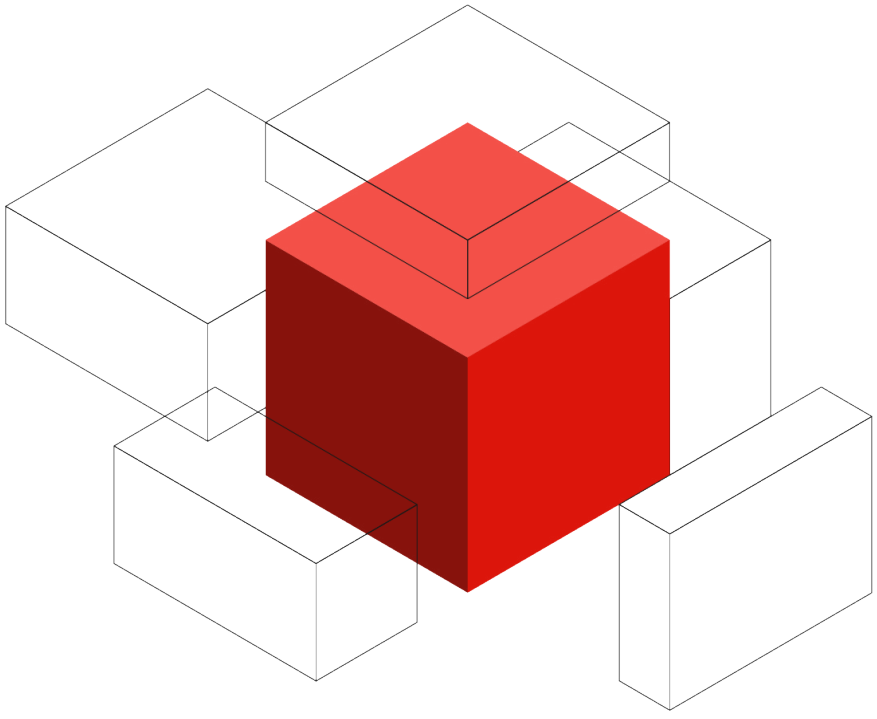
In the high-velocity world of SaaS, growth is both the goal and the challenge. As customer expectations rise and data volumes explode, success increasingly hinges on having the right data infrastructure—one that can scale elastically, deliver real-time insights, and maintain enterprise-grade reliability without inflating costs.

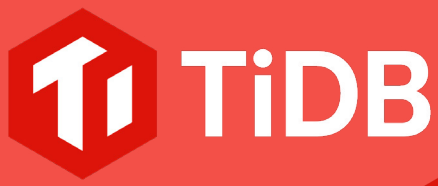
TiDB stands out as the proven, future-proof solution for multi-tenant SaaS platforms. Its distributed SQL architecture offers the horizontal scalability of NoSQL with the transactional integrity and SQL familiarity of traditional databases. With support for multi-tenancy, mixed workload processing, built-in disaster recovery, and robust security, TiDB enables SaaS companies to turn infrastructure into a growth enabler—not a bottleneck.

From startups launching new products to global enterprises modernizing legacy stacks, the path forward is clear: A scalable, performant, and intelligent data layer is no longer a luxury—it's a necessity. TiDB equips SaaS innovators with the tools they need to meet today's demands and build for tomorrow.

As the SaaS market continues to evolve, TiDB is ready to evolve with it—powering the next generation of platforms that are faster, smarter, and more resilient.

Ready to start your journey? Schedule a personalized tech talk today with one of our database experts to see how TiDB can power your next phase of SaaS growth.





EVALUATE TiDB FOR YOURSELF

Start Your Free Trial

Contact us for a personalized demo at pingcap.com/demo/