

How Practo Optimized Its Database Layer to Achieve 5X Performance and Lower AWS RDS Expenses



Overview

To support millions of patients every month, Practo required a highly stable and responsive platform. However, systemic lags and high operational maintenance overhead began to impact the patient booking experience and inflate infrastructure spending. By resolving bottlenecks and modernizing the database environment, Mydbops restored platform reliability, improved patient-doctor interaction speeds, and established a scalable foundation to support Practo's rapid global growth.



5X

Performance Boost

Accelerated database response speed through query tuning.

Reduced

AWS RDS Expenses

Slashed db maintenance costs by optimizing resource consumption.

Zero

Deployment Downtime

Enabled non-blocking db updates directly within the CI/CD pipeline.

24/7

Active DB Support

Established continuous database monitoring and preventative care.

RDS MySQL

Consulting Services

About Practo

Practo drives over 180 million annual patient interactions across a 400-million-user ecosystem. This massive engagement propelled the company's total GMV to ₹3,500 crore (\$411 million). Led by its recent US expansion, Practo's new AI marketplace alone contributed \$100 million to this surge. Consequently, the platform now commands a 70% market share in India while supporting 100,000 global healthcare providers.



We at Practo were looking for a full-time database admin and we met Mydbops team in the process. The decision to consult with Mydbops turned out to be one of the best decisions that we took. Mydbops team spent a week on 3 of our biggest MySQL databases and came up with a lot of suggestions. Once these suggestions were implemented, we gained a lot in our response times as well as more stable databases. Mydbops team has a lot of depth and insights into the working of MySQL and how it needs to be managed. It was a pleasure working with such a technically sound and dedicated team.

Hari Prashanth K R

General Manager Engineering, Practo



Deployment Type
Cloud-Based Deployment

Database Stack / Services Used
MySQL 5.1 to 5.5 & 5.6 on AWS RDS

Objective / Outcome
5X Performance Boost

Business Challenges

Overview

Practo needed a highly stable and responsive database platform that could support rapid application scaling and eliminate system-wide delays without driving up their cloud infrastructure budgets.

- Platform Slowness:** Inefficient, auto-generated database queries were causing high latency across the platform, directly impacting patient and doctor user experiences.
- Outdated Database Engine:** The platform was running on MySQL version 5.1, an outdated version that lacked modern optimization and performance capabilities.
- Engine Resource Bottlenecks:** The database suffered from performance limitations on AWS RDS temporary tables, which degraded processing speed during heavy transactional loads.
- Escalating Operational Costs:** Heavy and inefficient resource consumption by the database forced higher spending on AWS RDS maintenance and hosting, creating an unsustainable cost structure.

Goals

- Resolve database response lags to ensure stable, fast patient-doctor interactions.
- Upgrade the database engine from the outdated MySQL 5.1 to modern, supported versions.
- Minimize resource waste and lower monthly AWS RDS bills.
- Implement a reliable deployment workflow that avoids application downtime during database schema updates.

Solution Provided by Mydbops

Mydbops took on the challenge of restructuring Practo's database layer, implementing a multi-phased optimization strategy focused on platform reliability, user experience, and cost efficiency:

Peak Traffic Stabilization:

Resolved critical system slowdowns during high-volume booking hours by eliminating database disk constraints, ensuring patients and doctors experienced zero platform freezes.

Platform Modernization:

Upgraded the aging core database environment from MySQL 5.1 to MySQL 5.5, and subsequently to MySQL 5.6. This transition provided a modernized, highly stable infrastructure layer that eliminated platform lag and supported future traffic scaling.

Search and Booking Speed Optimization:

Replaced slow, auto-generated database calls with highly optimized, clean SQL queries, directly accelerating medical search results and booking confirmation times for patients.



Database Efficiency Overhaul:

Conducted a thorough schema and index refinement to clean the underlying database architecture, removing unnecessary constraints to lower operational overhead.

Infrastructure Value Maximization:

Optimized the database software to run in perfect alignment with the underlying cloud hardware, ensuring Practo extracted maximum efficiency and performance from their AWS cloud investments.

PLATFORM OPERATIONS BASELINE COMPARISON

Direct business outcomes mapped from database modernization interventions

× Legacy Bottlenecks

- Service Disruptions:** Database slowness directly stalled booking and search services during peak traffic windows.
- Infrastructure Waste:** Inefficient auto-generated queries drove heavy database resource consumption.
- Deployment Blockers:** Required database maintenance altered schema states with high risk of downtime.

✓ Modernized Infrastructure

- Platform Stability:** Relieved peak-traffic data load immediately, ensuring stable and reliable doctor discovery.
- Query Acceleration:** Deployed highly optimized, clean SQL queries to reduce resource footprint and cost.
- Non-Blocking Updates:** Enabled zero-downtime structural database alters integrated inside a modern CI/CD pipeline.

Results & Impact

Key Outcomes

✓ 5X Better Platform Performance:

Post-optimization, the database registered a 5X performance improvement, reducing response times and creating a highly stable database environment.

✓ Lower Infrastructure Expenses:

Optimizing queries and database performance directly minimized AWS RDS resource utilization, leading to lower monthly database bills.

✓ Non-Blocking CI/CD DB Deployments:

The transition to MySQL 5.6 allowed the platform to execute online alters. DB deployments were integrated into a modern CI/CD pipeline, ending application downtime during database schema updates.

✓ Constant Operational Peace of Mind:

Transitioned the database operations under Mydbops 24/7 continuous support, guaranteeing high availability and professional monitoring.

ENGINEERING & OPERATIONAL IMPACT

How database refactoring shifted internal development and operations from high friction to high efficiency

OPERATIONAL AREA	BEFORE MYDBOPS OPTIMIZATION	WITH MYDBOPS DATABASE MANAGEMENT
Application Deployments	Blocked Pipeline Database schema alterations were blocked and risked application downtime.	Continuous Delivery Automated, non-blocking online alters natively built inside the CI/CD pipeline.
Infrastructure Overhead	Manual Monitoring Development teams regularly had to diagnose database slowdowns and manual fixes.	Fully Managed Support Ongoing proactive health monitoring covered entirely under 24/7 expert database care.
Query Lifecycle	Resource Heavy Inefficient, auto-generated database queries consistently exhausted disk resources.	Optimized SQL Clean, structured SQL queries executing with lightweight hardware utilization.

Building a Scalable Database Foundation

Digital healthcare platforms operate under strict demands for uptime and responsiveness. For a patient looking to book an appointment or consult a doctor, even a few seconds of delay can disrupt critical care coordination. Practo faced a crossroads where its rapid business growth collided with the technical limitations of an aging database environment. Heavy resource consumption, sluggish query execution, and high cloud costs threatened to impact service quality.

By partnering with Mydbops, Practo did not just receive a quick fix. The collaboration started by resolving the immediate disk IO bottlenecks using provisioned IOPS, followed by a systematic overhaul of query structures, index designs, and engine configurations. Upgrading to newer database engines allowed Practo to run a leaner, faster, and more cost-efficient database stack on AWS RDS.

The final step of building online database alterations directly into a CI/CD pipeline meant the engineering team could deploy new features without interrupting patient access. This technical turnaround established a highly stable database layer capable of supporting Practo's journey into global markets and massive user scales.

[Contact Our Database Experts Today](#)