

FAST CARS & FASTER

APIS FOR DISRUPTIVE CAR

RENTAL PROVIDER

CAR RENTAL GATEWAY

- Industry: Information Technology
- Location: UK
- Data Center: 2 (Amazon Web Services)

USE CASE

A platform provider for the car rental industry requiring scale and performance for their application

WHY SEVERALNINES

Simplified deployment, multi-technology support, and performance monitoring to ensure uptime and scale.

INTRODUCTION

Launched in 2014, Car Rental Gateway is turning the car rental industry upside-down by providing everyone access at reasonable rates to cutting edge technology that only the largest distributors can afford otherwise. Their platform provides a high-traffic API designed to allow rental car brokers and suppliers to access the fleets of over a hundred rental car suppliers. In return, suppliers are able to provide their fleet of cars for booking through the system.

The platform operates on a SaaS model and handles 10s of millions of API calls per day. This means instant technological capability to directly connect to over 150 online car rental suppliers, from large global brands to top regional and local car rental companies. In addition to connectivity, the platform offers up to date unified multi-lingual supplier content and support multiple currencies and payment methods. Security is a key aspect of the platform, which is PCI-DSS compliant.

Car Rental Gateway is disrupting the market, offering functionality unthinkable in GDSs (Global Distribution Systems) as it allows its customers to freely to build their own client base and use their own independent customized branding. The platform is currently in its third generation of maturity.

The team behind Car Rental Gateway has been steering e-commerce solutions in car rental brokering for over 15 years with experience working with top global players like Rentalcars.com and Expedia.

“We needed a **distributed database technology** that was a bit more **resilient to network issues** and a system that could provide the **setup, management and monitoring capabilities** of the **database cluster**”

Gotthard Schlüter, Lead DevOps Engineer

CHALLENGE

The team making sure the platform runs at peak performance consists of System Administrators and DevOps Engineers. At the time of writing, they handle more than 100 million API requests a day across some 150 AWS instances. The database technology powering this service used MySQL Galera Cluster which, unfortunately, was not performing well or allowing them to scale efficiently.

Before starting a search for a new way, the team tried to perfect their Galera setup. “Our Galera Cluster had, at one point, up to ten nodes,” said Gotthard Schlüter, Lead DevOps Engineer at Car Rental Gateway. “And we were using post-failover scripts across three different data centers. At that level, Galera became susceptible to network issues.” With levels of over a 1000 inserts per second, any network instability would wreak havoc with the system performance and data integrity.

The team eventually made the strategic decision to move away from the synchronous replication technology offered by Galera to a more traditional MySQL-like, replication setup (specifically a master-slave

asynchronous cluster). In the end the team decided to utilize MariaDB at their technology of choice.

In addition, the team felt they did not have the right tools in place to monitor and scale their new MariaDB setup and turned to the web to find a database management solution to ensure their deployments were deployed for HA and that they had the ability to closely monitor and tweak performance. "We needed a distributed database technology that was a bit more resilient to network issues," said Schlüter. "And a system that could provide the setup, management and monitoring capabilities of the cluster."

SOLUTION

Initially, for safety and risk mitigation reasons, the CRG team leveraged ClusterControl for the existing Galera Cluster setup while developing the plan for the switch-over to asynchronous replication with MariaDB.

While using ClusterControl for Galera, the CRG team configured and deployed a MariaDB replication cluster with eight slave nodes and two additional nodes utilized for backups and system analytics. The cluster sits behind two Amazon AWS Elastic Load Balancers (the Master behind one and the slaves behind the other). The team also leveraged ClusterControl's pre- and post-failover/switchover hooks. The team developed custom scripts that could be hooked into ClusterControl to integrate database failover with the overall infrastructure failover strategy. For instance, in case of master failure, new nodes are spun up and ELBs are updated. This, in turn, fundamentally automates the ability to scale up or down based on the needs of the application.

"In two months we were able to go live with new MariaDB replication clusters using ClusterControl," said Schlüter. "We did not even need assistance from Severalnines to get into production."

NEXT STEPS

Car Rental Gateway is also operating several MongoDB databases and plans to explore a potential expansion of ClusterControl to bring their different open source databases into one platform allowing for consistent management.

WHY SEVERALNINES?

ClusterControl provided Car Rental Gateway with several features they required to take their platform to the next level.

- **Simplified Cluster Setup & Management:** The point-and-click interface and predefined configuration templates allowed CRG to quickly spin up new environments. Even in production, the ease of adding new slaves or rebuilding corrupted slaves has been a benefit. "The simple mechanism of adding database nodes into the cluster is a real time-saver," said Schlüter.
- **Performance Monitoring:** ClusterControl provides both high-level and deep-dive monitoring of all aspects of the database stack. These "health checks" allowed the CRG team to identify and tweak specific issues in their setup to keep their platform engine humming.
- **Scale & Expanded Support:** ClusterControl allowed the team the ability to experiment with several different database and load balancing technologies before making a final decision. It also offers the ability to manage multiple database types alongside one another, like CRG had to do when running Galera while experimenting with MariaDB.

