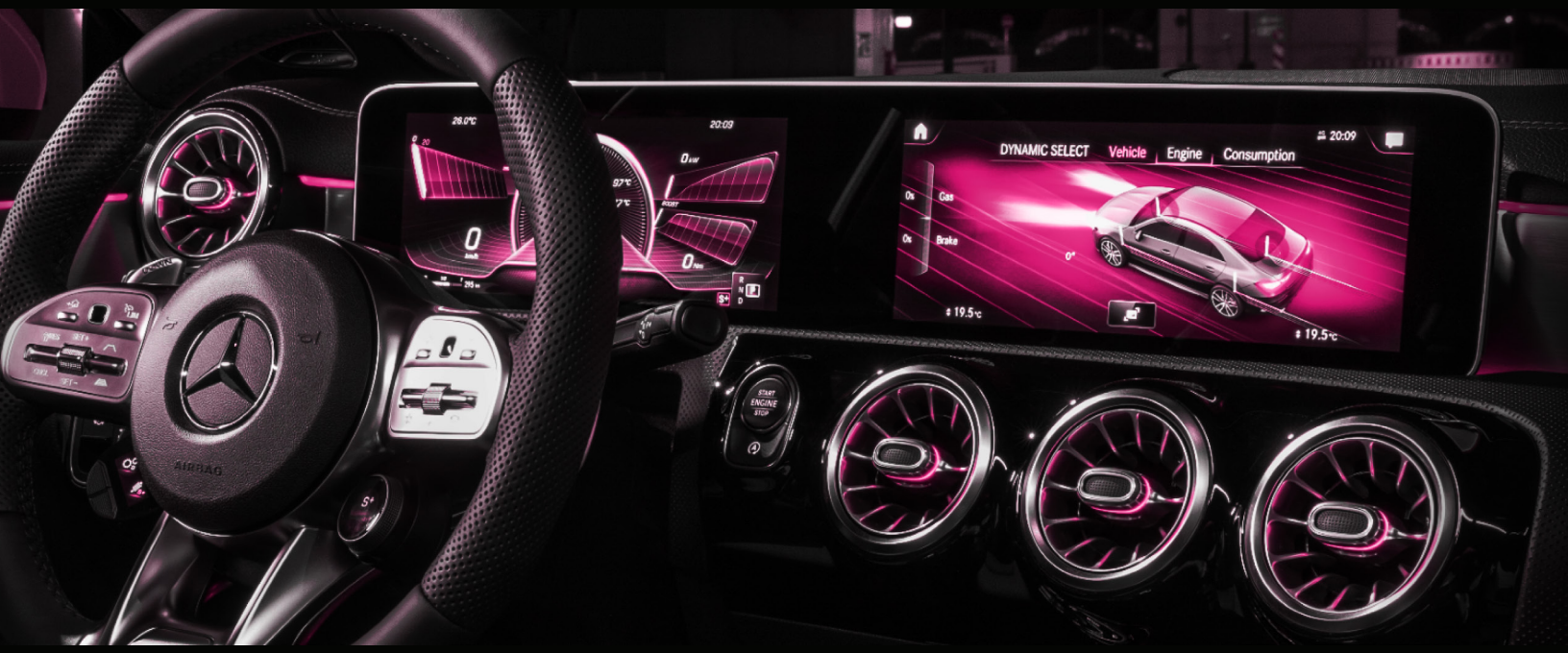
HashiCorp



Mercedes-Benz | CONSUL CASE STUDY

On the road again

Cars, Kubernetes, and HashiCorp Consul. How Mercedes-Benz delivers on service networking to accelerate delivery of its next-gen connected vehicles..

// Infrastructure Enables Innovation

Mercedes-Benz Summary

Mercedes-Benz Research and Development North America (MBRDNA) develops the world's most advanced automotive technology and vehicle design with luxury and style. It's not just about cars at MBRDNA, it's also about the latest and greatest software, cutting-edge technology, and groundbreaking innovation. Devoted to rethinking the automobile, MBRDNA focuses on creating the next generations of connected, autonomous, electric vehicles and the ecosystems enabling them.

MERCEDES-BENZ FAST FACTS

25 years of technical innovation at Mercedes-Benz

< 30 min to request & deploy new Kubernetes cluster

Connecting & coordinating **billions** of data points

+200 services on several Kubernetes clusters

From **days to minutes** to discover and connect services

Proof of concept to production in only **4 weeks**

The best or nothing

> The visibility, transparency, and control we have with Consul eliminates so many of the service discovery and connectivity obstacles that used to prevent us from working as quickly and efficiently as we wanted.

As the inventor of the automobile, Mercedes-Benz, a brand of the Daimler Group, has pushed the limits of the concept of an automobile, from introducing the first multi-valve engine to setting new standards in safety testing and performance engineering. But constantly testing the limits of automotive technology takes more than just imagination and a love of driving.

It requires being able to combine modern computing technologies with advanced data modeling capabilities to achieve new innovations and breakthroughs. And in the age of connected devices and autonomous vehicles, it requires building a robust and agile cloud infrastructure capable of connecting and coordinating the billions of data points that will form the 'brain' that drives its connectivity, a job well suited for Mercedes-Benz Research and Development North America.

"Daimler created a research and development hub specifically to build the foundation for its connected and autonomous vehicle portfolio under the Mercedes brand," Sriram Govindarajan, principal infrastructure engineer at Mercedes-Benz Research & Development (MBRDNA) says. "Almost immediately, we recognized that in order to build products of tomorrow we had to first build an IT infrastructure suitable for today, which meant moving the majority of our existing infrastructure to the cloud and modernizing our development processes at the same time."

Challenges

Transitioning from on-premises to cloud and kubernetes infrastructure

Simplifying and accelerating service discovery and networking

Improving efficiency and productivity while controlling costs

> The visibility, transparency, and control we have with Consul eliminates so many of the service discovery and connectivity obstacles that used to prevent us from working as quickly and efficiently as we wanted.

Zero visibility stifles performance

Connected cars have raced to the forefront of the Internet of Things (IoT) trend, designed to share data in real-time with other systems outside the vehicle to dramatically improve safety, performance, and passenger comfort. Each aspect of that connectivity is powered by a purpose-built app, and each app is out powered by an array of microservices that must connect with other services to complete the system.

With Microsoft Azure as its cloud platform, MBRDNA adopted Kubernetes to containerize development and provide the speed and agility required to deliver new features faster. "Kubernetes was ideal for us because it centralizes the code base, infrastructure, and everything else our teams need to quickly develop the features and capabilities in our roadmap," Govindarajan says. "But the more services we ran in each cluster, the harder it became to match them up and connect them because we didn't have the right service discovery mechanisms in place."

Govindarajan says that development teams may have services running in their individual clusters and had no way of identifying other services or dependent services running on other clusters and given how quickly service addresses, hosts, and ports are added and changed, doing so without a dedicated service discovery tool was virtually impossible. "Service discovery is 100 percent essential to our entire operation and to the ability to meet our responsibilities to the broader organization," he says. "A developer could scour our entire environment for days and never find what he or she was looking for, so we needed a way to accelerate discovery and do it effortlessly across a number of clusters and clouds."

> Consul lets us spread more than 200 microservices over several AKS clusters. Each AKS cluster connects to a local Consul client, which feeds into a Consul cluster that forms a larger service discovery mesh that allows us to find and connect services in a matter of minutes with minimal effort.

SRIRAM GOVINDARAJAN, PRINCIPAL INFRASTRUCTURE ENGINEER,
MERCEDES-BENZ RESEARCH & DEVELOPMENT (MBRDNA)

Cloud-based, centralized service discovery for greater efficiency

After briefly evaluating both open source and paid service discovery tools, MBRDNA chose HashiCorp Consul as their Service Discovery mechanism in their effort to migrate to Azure.

With the initial architecture in place, Govindarajan's team focused its attention on their critical work: optimizing its code development operations.

nlike the other on-premises and open source service discovery tools MBRDNA evaluated, cloud native Consul features dynamic application and infrastructure service location and easy connection across any runtime platform or cloud. The centralized services registry, automated centralized network middleware configuration, and real-time directory of all running services combine to dramatically improve application inventory management and faster service connectivity.

For example, MBRDNA developers can now request an Azure Kubernetes Service (AKS) cluster through a self-service portal and have everything they need for their deployment — code, scripts, APIs, and Consul clients — in less than 30 minutes. Once their services have been deployed, other teams can simply use code or scripts on their clusters to look up the services or the Consul client they need and connect them instantly.

"Consul lets us spread more than 200 microservices over several AKS clusters," Govindarajan explains. "Each AKS cluster connects to a local Consul client, which feeds into a Consul cluster that forms a larger service discovery mesh that allows us to find and connect services in a matter of minutes with minimal effort."

Life in the fast lane

Govindarajan says that while it can be hard to quantify specific improvements, HashiCorp solutions have fundamentally changed the way his team works.

"Consul and the other HashiCorp tools allowed us to bring the entire development process in-house and stop relying on third-parties to own a portion of our workflows," he says. "The visibility, transparency, and control we have with Consul eliminates so many of the service discovery and connectivity obstacles that used to prevent us from working as quickly and efficiently as we wanted."

The intuitive design and on-demand support resources allowed MBRDNA to configure and launch Consul in just 12 weeks, despite how new the team was to the world of DevOps and cloud infrastructure. As the team became more familiar and comfortable with Consul, they began using the product's other efficiency-oriented features like server queries and service segmentation to further reduce the strain on both the company's networks and its budget.

Rather than procuring different servers for production, staging, and development environments, the team uses tags that indicate whether the target running on a central Consul server is in production, development, or staging. This method simplifies discovery and cuts down on unnecessary client-cluster connections, while also significantly reducing certificate and server maintenance costs.

Business outcomes

- **Automated discovery of more than 200 services across several Kubernetes clusters**
- **Created a Consul-based service discovery platform for greater visibility and transparency**
- **Transitioned entire development process in-house, away from third-party outsource vendors**
- **Reduced certificate and server maintenance costs using network tags**

Conclusion

Govindarajan says that his team's success to this point inspires confidence for how they'll use it in the future. "Consul proved to be the right solution for solving our service discovery challenges that we found at the right time," he says. "We're eager to start using the upgraded features in the next versions of Consul to continue strengthening the resilience and performance of our infrastructure and help usher in the next generation of innovative driving machines."

MBRDNA Partner

Based in the Pacific Northwest, Sriram has more than 20 years of engineering experience in the IT industry. With a heavy focus on the DevOps philosophy, Sriram is very well versed in container orchestration with Kubernetes and embodies a strong understanding of cloud architecture. Currently Sriram is Principal Infrastructure engineer at MBRDNA, where he works on its Autonomous and Connected programs within the Mercedes-Benz Innovation umbrella.

Sriram Govindarajan, Principal Infrastructure Engineer, Mercedes-Benz Research & Development

Technology Stack

- Infrastructure: 100% on Azure
- Platform: Majority Containers and some VMs
- Proxies: Planned for future to support service mesh
- Load balancers: Azure public load balancers
- Firewalls: Azure Firewalls
- API gateway: Azure APIM
- CA: Internal Daimler provided
- IAM: AAD and OAuth based internal tooling
- APM: Mostly Azure Application Insights and some AppDynamics
- Provisioning: Terraform Providers
- Security management: Azure Key Vault, transitioning to HashiCorp Vault


HashiCorp