



CONSUL-FALLSTUDIE

Wieder auf Achse

Connected Cars, Kubernetes und HashiCorp Consul: Wie Mercedes-Benz mit intelligenten Netzwerkservices die Entwicklung seiner Next-Generation-Fahrzeuge beschleunigt.

// Innovation durch Infrastruktur

Kurzinfo zu MBRDNA

Mercedes-Benz Research and Development North America (MBRDNA) entwickelt die innovativsten Automobiltechnologien und Fahrzeugdesigns der Welt, luxuriös und stilvoll zugleich. Dabei geht es nicht nur um Autos, sondern auch um die neueste und beste Software, um Spitzentechnologien und um bahnbrechende Innovationen. MBRDNA denkt die Automobilität neu und konzentriert sich auf die Entwicklung der kommenden Generationen vernetzter, autonomer Elektrofahrzeuge und der dazu nötigen Systemumfelder.

FAST FACTS



25 Jahre technische Innovation bei Mercedes-Benz



< 30 Minuten von der Anforderung bis zum Deployment eines neuen Kubernetes-Clusters



Verknüpfung und Orchestrierung von **Milliarden** von Datenpunkten



+200 Services auf mehreren Kubernetes-Clustern



Service Discovery und Vernetzung **in Minuten** statt in Tagen



Vom Proof of Concept zur Produktion in nur **vier Wochen**

Das Beste oder gar nichts

- // Mit der Sichtbarkeit, Transparenz und Kontrolle, die uns Consul gibt, werden viele der Service-Discovery- und Konnektivitäts-hindernisse beseitigt, die uns früher daran gehindert haben, so schnell und effizient zu arbeiten, wie wir wollen.

Mercedes-Benz, Erfinder des Automobils und heute eine Marke des Daimler-Konzerns, hat das Automobil immer wieder neu definiert, von der Einführung des ersten Mehrventilmotors bis hin zur Setzung neuer Maßstäben in der Sicherheitsprüfung und Leistungstechnik. Will man jedoch die Grenzen der Automobiltechnik immer wieder aufs Neue austesten, braucht es weit mehr als Fantasie und Freude am Fahren.

Wenn man Innovationen und immer wieder Durchbrüche erreichen möchte, muss man moderne Computertechnologie mit fortschrittlicher Datenmodellierung verbinden. Im Zeitalter vernetzter Geräte und autonomer Fahrzeuge braucht es außerdem robuste und agile Cloud-Infrastrukturen, die in der Lage sind, die Milliarden von Datenpunkten miteinander zu verbinden und zu koordinieren – gewissermaßen das „Gehirn“, die diese Konnektivität ausmachen. Eine Aufgabe, die wie gemacht ist für Mercedes-Benz Research and Development North America.

„Daimler hat ein Forschungs- und Entwicklungszentrum geschaffen, um die Basis für sein Portfolio an vernetzten und autonomen Fahrzeugen unter der Marke Mercedes zu legen“, erläutert Sriram Govindarajan, leitender Ingenieur für Infrastrukturfragen bei Mercedes-Benz Research & Development (MBRDNA). „Dabei haben wir ziemlich früh erkannt, dass wir, um Produkte von morgen zu bauen, zunächst eine IT-Infrastruktur für heute aufbauen müssen. Das bedeutete, dass ein Großteil unserer bestehenden Infrastruktur in die Cloud verlagert werden musste und dass wir gleichzeitig unsere Entwicklungsprozesse modernisierten.“

Herausforderungen



Von einer On-premises-Infrastruktur auf Cloud und Kubernetes umsteigen



Service Discovery und Verbindungen vereinfachen und beschleunigen



Effizienz und Produktivität verbessern – bei gleichzeitiger Kostenkontrolle

- //** Kubernetes war perfekt, weil es die Codebasis, die Infrastruktur und alles übrige zentralisiert, was wir für die schnelle Entwicklung brauchen. Aber je mehr Dienste in den Clustern liefen, desto schwieriger wurde es, sie zuzuordnen und miteinander zu vernetzen.

Ohne Sichtbarkeit keine Leistung

Vernetzte Fahrzeuge führen das IoT-Feld (Internet of Things) an. Sie können Sicherheit, Leistung und Fahrkomfort wesentlich verbessern, indem sie in Echtzeit Daten mit externen Systemen austauschen. Hinter jeder Funktion dieser Konnektivität steht eine speziell entwickelte Anwendung, und hinter jeder Anwendung eine Vielzahl von Microservices, die sich mit weiteren Diensten verbinden, damit das System funktioniert.

MBRDNA hat als Cloud-Plattform Microsoft Azure gewählt, Kubernetes eingeführt und damit die Entwicklung containerisiert, sodass sie nun die für die Bereitstellung neuer Features nötige Geschwindigkeit und Agilität bietet. „Kubernetes war perfekt, weil es die Codebasis, die Infrastruktur und alles übrige zentralisiert, was wir für die schnelle Entwicklung brauchen“, erklärt Govindarajan. „Aber je mehr Dienste in den Clustern liefen, desto schwieriger wurde es, sie zuzuordnen und miteinander zu vernetzen.“

Govindarajan erklärt das so, dass die Entwicklerteams zum Teil Services in ihren Clustern laufen hatten, aber keine Möglichkeit sahen, andere Services oder abhängige Services in anderen Clustern zu identifizieren. Weil Adressen, Hosts und Ports extrem schnell neu hinzukommen und sich ändern, war das ohne ein dediziertes Service Discovery Tool auch praktisch unmöglich: „Service Discovery ist für unsere gesamte Operation unverzichtbar – und dafür, dass wir unserer Verantwortung gegenüber dem Gesamtunternehmen gerecht werden“, sagt er. „Entwickler konnten tagelang unsere gesamte Umgebung durchkämmen, ohne zu finden, wonach sie suchten. Wir brauchten also eine Möglichkeit, die Discovery zu beschleunigen, und zwar möglichst unkompliziert über mehrere Cluster und Clouds hinweg.“

// Mit Consul können wir über 200 Microservices auf mehrere AKS-Cluster verteilen. Jedes AKS-Cluster hat einen lokalen Consul-Client, der in ein Consul-Cluster einspeist, das zu einem großen Service-Mesh gehört. So können wir Services in wenigen Minuten mit minimalem Aufwand finden und verbinden.

SRIRAM GOVINDARAJAN, LEITENDER INGENIEUR FÜR INFRASTRUKTURFRAGEN,
MERCEDES-BENZ RESEARCH & DEVELOPMENT (MBRDNA)

Mehr Effizienz durch cloudbasierte, zentralisierte Service Discovery

Nach einer kurzen Evaluierung von Open-Source- und von kostenpflichtigen Service-Discovery-Tools entschied sich MBRDNA bei der Migration zu Azure für die Service-Discovery-Lösung Consul von HashiCorp.

Nachdem die grundlegende Architektur geschaffen war, konzentrierte sich Govindarajans Team auf seine wesentliche Arbeit: die Optimierung der Code-Entwicklung.

Im Gegensatz zu den anderen On-premises- und Open-Source-Service Discovery-Tools, die MBRDNA untersucht hat, punktet das cloud-native Consul mit einer dynamischen Erkennung von Anwendungs- und Infrastrukturdiensten sowie mit einfacher Vernetzung über alle Laufzeitplattformen und Clouds hinweg. Die zentralisierte Services Registry, die automatisierte zentralisierte Netzwerk-Middleware-Konfiguration und die Echtzeitauflistung aller laufenden Services ermöglichen eine erheblich verbesserte Bestandsverwaltung der Anwendungen und schnellere Service-Konnektivität.

Die Entwickler von MBRDNA können jetzt beispielsweise über ein Selfservice-Portal einen AKS-Cluster (Azure Kubernetes Service) anfordern und haben in weniger als 30 Minuten alles zur Hand, was sie für das Deployment benötigen: Code, Skripte, APIs und Consul-Clients. Sobald die Services laufen, können andere Teams den Code oder die Skripte ganz einfach in ihren Clustern verwenden, um die Services oder den Consul-Client, den sie benötigen, nachzuschlagen und sofort zu verbinden.

„Mit Consul können wir über 200 Microservices auf mehrere AKS-Cluster verteilen“, erklärt Govindarajan. Jedes AKS-Cluster hat einen lokalen Consul-Client, der in ein Consul-Cluster einspeist, das zu einem großen Service-Mesh gehört. So können wir Services in wenigen Minuten mit minimalem Aufwand finden und verbinden.“

Arbeit auf der Überholspur

Govindarajan zufolge kann man die einzelnen Verbesserungen zwar kaum beziffern, doch insgesamt, sagt er, haben die Lösungen von HashiCorp die Arbeitsweise seines Teams grundlegend verändert.

„Mit Consul und den anderen Tools von HashiCorp konnten wir den gesamten Entwicklungsprozess inhouse holen, und wir müssen uns nun nicht mehr auf Dritte verlassen, in deren Händen ein Teil unserer Arbeitsabläufe liegt“, erläutert er. „Mit der Sichtbarkeit, Transparenz und Kontrolle, die uns Consul gibt, werden viele der Service-Discovery- und Konnektivitätshindernisse beseitigt, die uns früher daran gehindert haben, so schnell und effizient zu arbeiten, wie wir es wollen.“

Durch das intuitive Design und die On-Demand-Support-Ressourcen konnte MBRDNA Consul in nur zwölf Wochen konfigurieren und starten – obwohl das Team in der Welt von DevOps und Clouds noch recht neu war. Als das Team dann im Umgang mit Consul geübter und sicherer wurde, nutzte es auch die weiteren effizienzorientierten Features des Produkts, etwa Serverabfragen und Service-Segmentierung, um das Unternehmensnetzwerk zu entlasten – und damit auch das Budget.

Anstatt unterschiedliche Server für Produktions-, Staging- und Entwicklungsumgebungen bereitzustellen, verwendet das Team Tags, die anzeigen, ob sich das Ziel, das auf einem zentralen Consul-Server läuft, in Produktion, Entwicklung oder Staging befindet. Das erleichtert die Discovery und reduziert die Zahl unnötiger Client-Cluster-Verbindungen, ebenso wie die Kosten für Zertifikate und die Serverwartung.

Geschäftserfolge



Automatisierte Discovery von über 200 Services über mehrere Kubernetes-Cluster



Aufbau einer Service-Discovery-Plattform mit Consul, die für mehr Sichtbarkeit und Transparenz sorgt



Umstellung des gesamten Inhouse-Entwicklungsprozesses, weg von externen Drittanbietern



Kostensenkung bei Zertifikaten und Serverwartung durch Netzwerktags

Fazit

Govindarajan betont, dass die Erfolge, die sein Team bis hierhin erzielt hat, das Vertrauen hinsichtlich der künftigen Nutzung gestärkt haben. „Consul hat sich als die richtige Lösung für unsere Herausforderungen im Bereich Service Discovery erwiesen, und wir haben sie genau zum richtigen Zeitpunkt gefunden“, fügt er hinzu. „Wir freuen uns auf die Feature-Upgrades der nächsten Versionen von Consul, mit denen wir die Widerstandsfähigkeit und die Leistungsfähigkeit unserer Infrastruktur weiter stärken wollen – und damit den Weg für die nächste Generation innovativer Antriebsmaschinen ebnen.“

Partner bei MBRDNA



Sriram Govindarajan lebt im pazifischen Nordwesten der USA. Er hat über 20 Jahre Erfahrung als IT-Ingenieur. Schwerpunkte seiner Arbeit sind DevOps, Container-Orchestrierung mit Kubernetes und Cloud-Architekturen. Sriram Govindarajan ist derzeit bei MBRDNA als leitender Ingenieur für Infrastrukturfragen tätig, wo er unter dem Dach von Mercedes-Benz Innovation an autonomen Fahrzeugen und Connected Cars arbeitet.

Sriram Govindarajan, Leitender Ingenieur für Infrastrukturfragen,
Mercedes-Benz Research & Development

Technologie-Stack

- Infrastruktur: zu 100% auf Azure
- Plattform: hauptsächlich Container sowie ein paar VMs
- Proxies: sollen künftig das Service-Mesh unterstützen
- Lastverteilung: öffentlicher Load Balancer von Azure
- Firewalls: Azure
- API-Gateway: APIM von Azure
- Zertifizierungsstelle: intern durch Daimler
- IAM: AAD- und OAuth-basierte interne Werkzeuge
- APM: Hauptsächlich Application Insights von Azure und etwas AppDynamics
- Provisioning: mithilfe von Terraform
- Sicherheitsmanagement: Azure Key Vault, Umstellung auf HashiCorp Vault

