

Case Study: Vom Systemdschungel zur produktzentrierten Enterprise-Sicht

Architektur als aktiver Teil der Produktorganisation

30
JUN
2026

Philip Daubmeier

Enterprise Architect Connected Car & Cloud bei CARIAD



CARIAD entwickelt für den Volkswagen Group Software Stack **synergetische Produkte** für Millionen Fahrzeuge.



Group Driver Stacks

Group Experience Stacks

Group Cloud Stacks

Group Motion Stacks

CLOUD

Wir verwandeln das Auto in das ultimative vernetzte Gerät

ONE Group Cloud Plattform

In-house Fahrzeugbackend, Cloud Plattform und Ökosystem für alle Volkswagen Konzern Marken

9 AUTOMOBIL MARKEN



45+

Millionen
Fahrzeuge verbunden

24/7

365 Tage globaler Betrieb

80+

Globale Märkte
mit Onlinediensten

Connectivity



Ermöglicht die Echtzeit-Interaktion zwischen Fahrzeugen, Nutzern, Infrastruktur und Diensten (V2X) für mehr Sicherheit, Verkehrsoptimierung, globale Konnektivität und Remote-Updates mittels eSIM.

Cloud Services



Ermöglicht Fahrzeugdienste wie We Connect, We Charge sowie OTA-Updates für alle Marken und Fahrzeugplattformen – für eine schnelle Markteinführung und hohe Wiederverwendbarkeit.

Data Platform

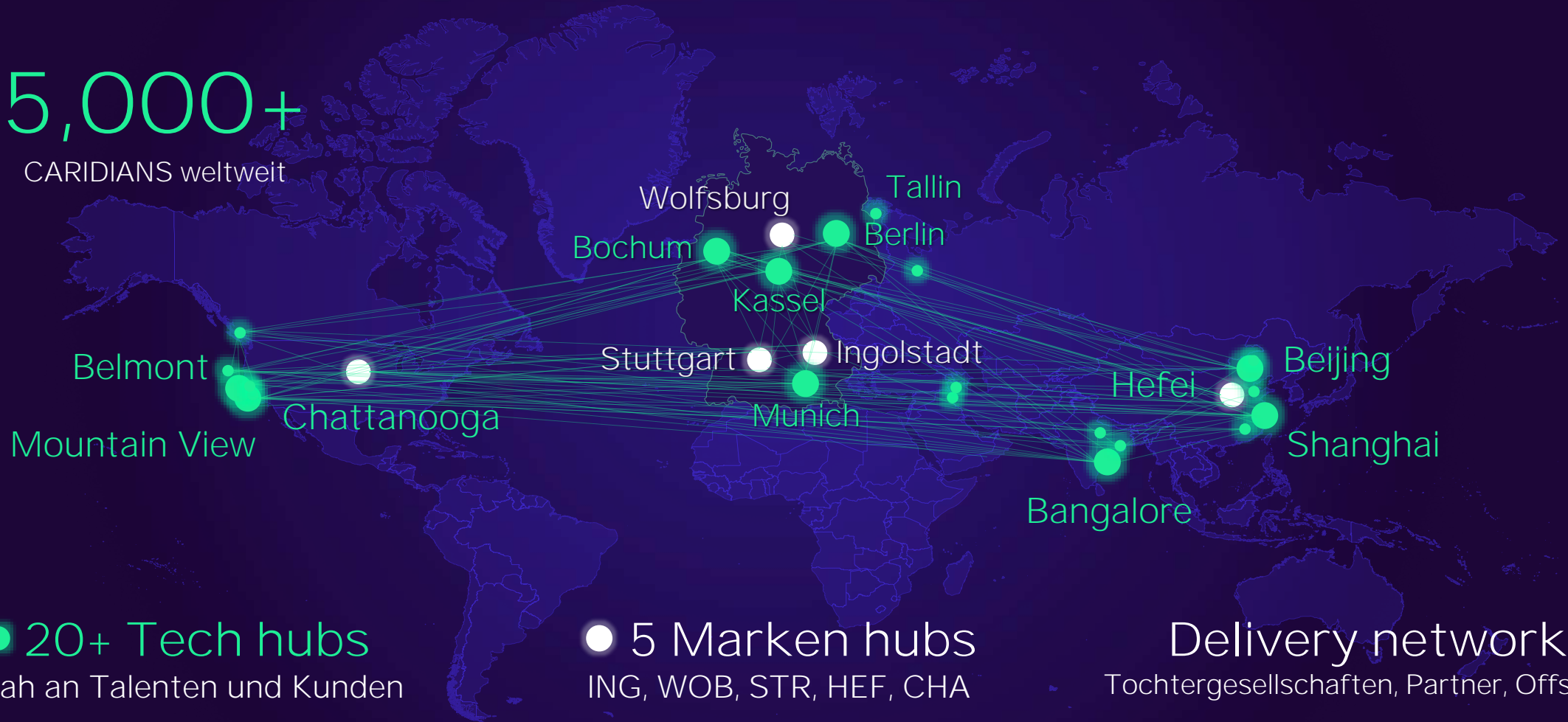


Führt Fahrzeugdaten aus dem gesamten Volkswagen Konzern in einem einheitlichen Datenökosystem zusammen und unterstützt so datengetriebene Entwicklung, Anomalieerkennung sowie Flottenanalysen.

CARIAD ist global aufgestellt und nah an Innovationen, Talenten und Kunden

5,000+

CARIDIANS weltweit



● 20+ Tech hubs
Nah an Talenten und Kunden

● 5 Marken hubs
ING, WOB, STR, HEF, CHA

Delivery network
Tochtergesellschaften, Partner, Offshore

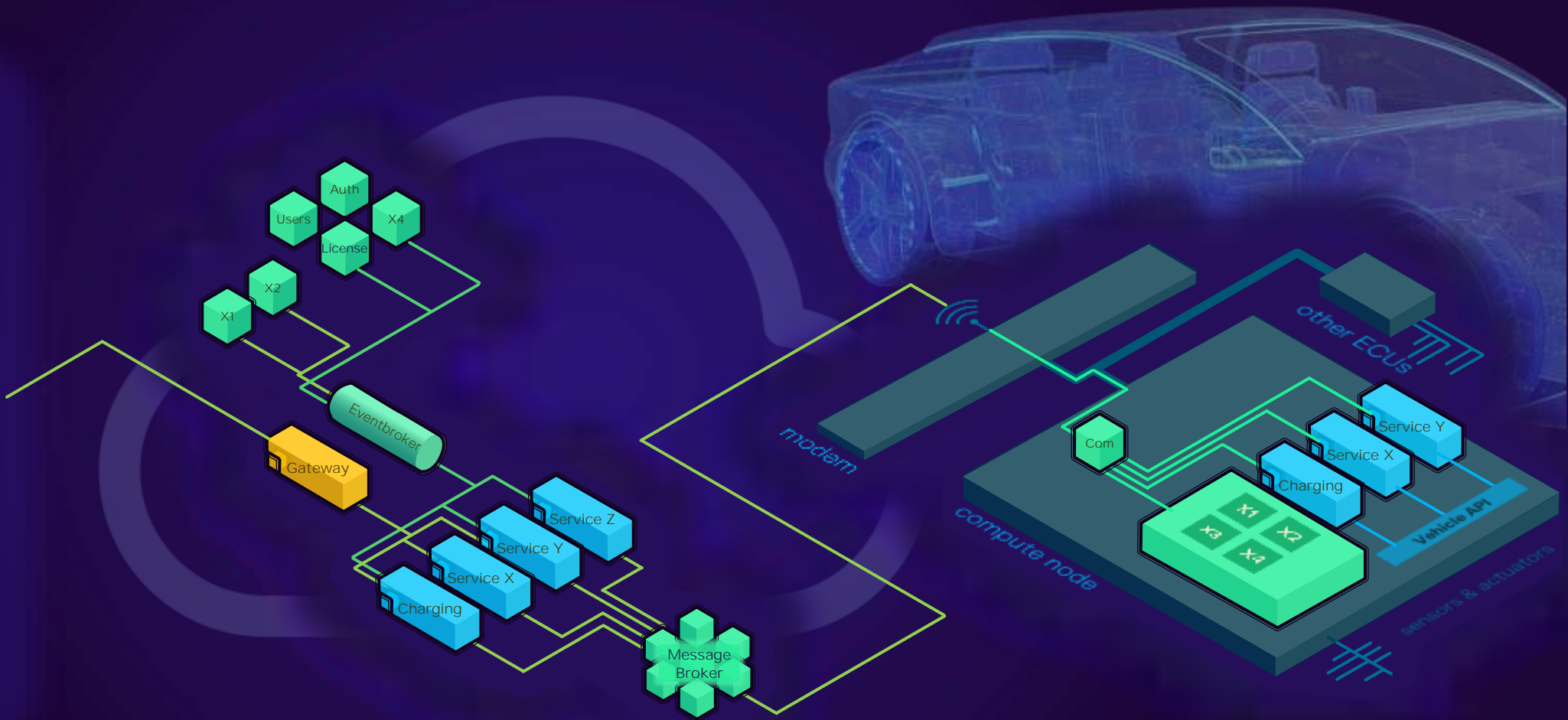


C A R I A D

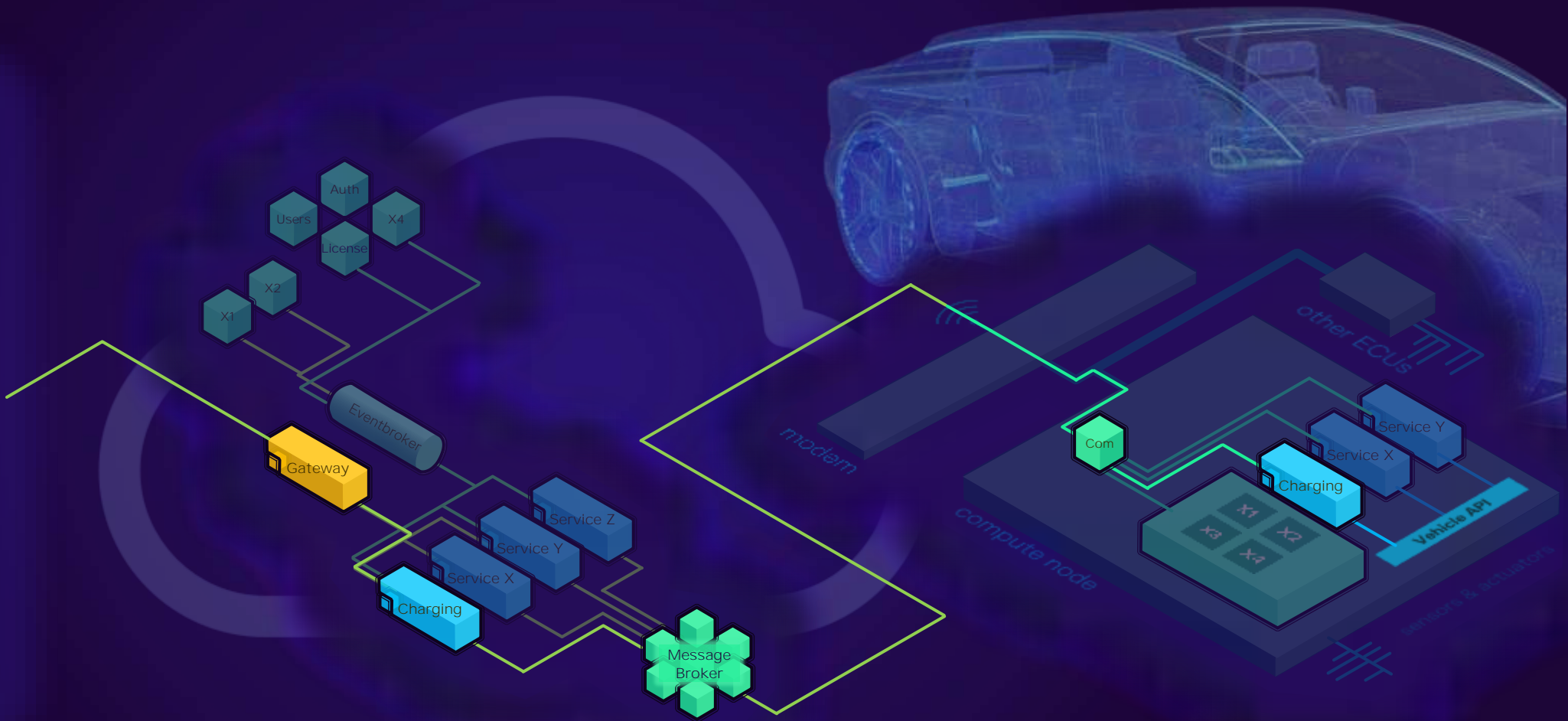
Beispiel: Ladevorgang per App steuern



Beispiel: Ladevorgang per App steuern



Beispiel: Ladevorgang per App steuern



Jedes Team hatte seine eigene Wahrheit

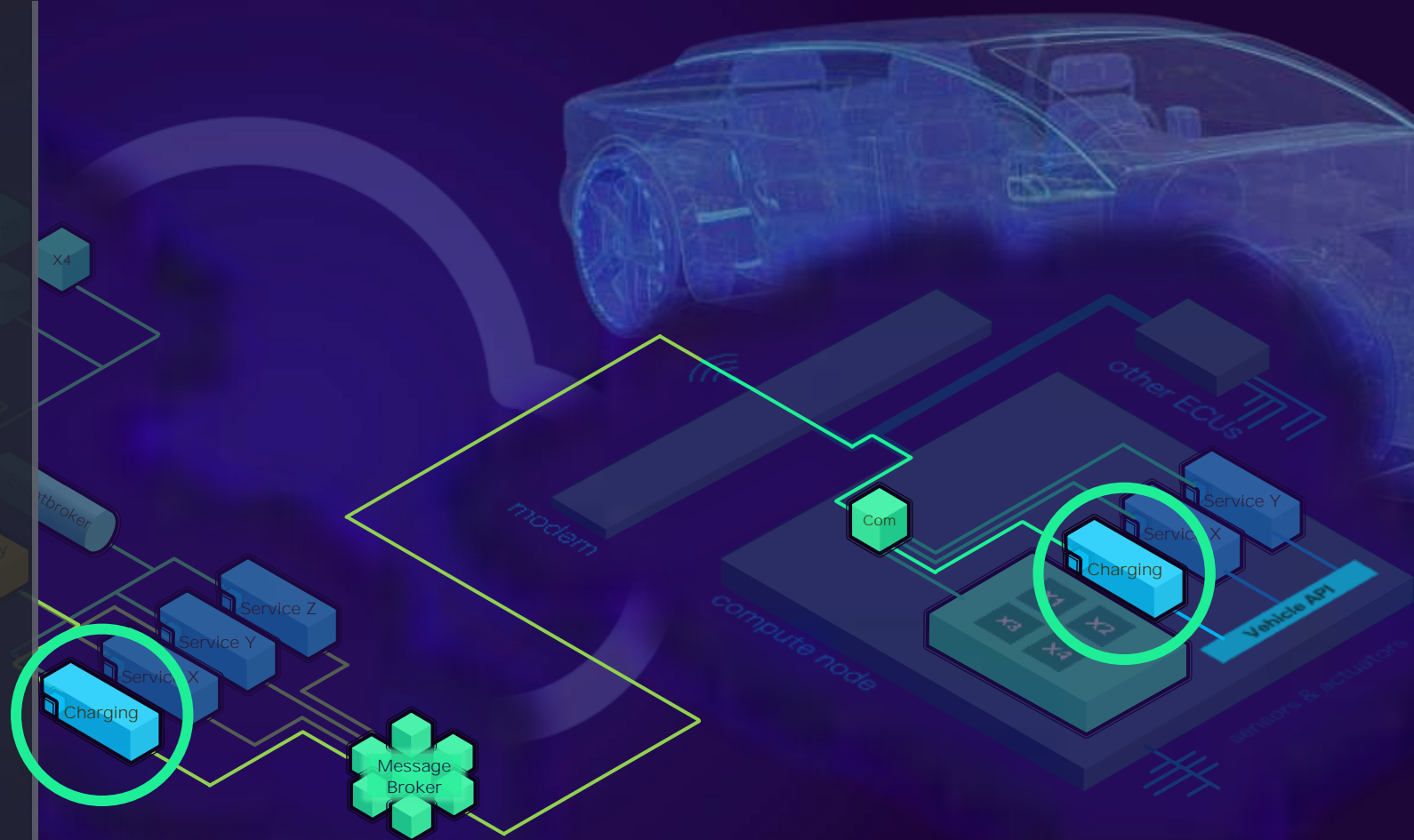
Gleiche Namen für unterschiedliche Systeme

- // „Charging Service“ im Fahrzeug
- // „Charging Service“ in der Cloud

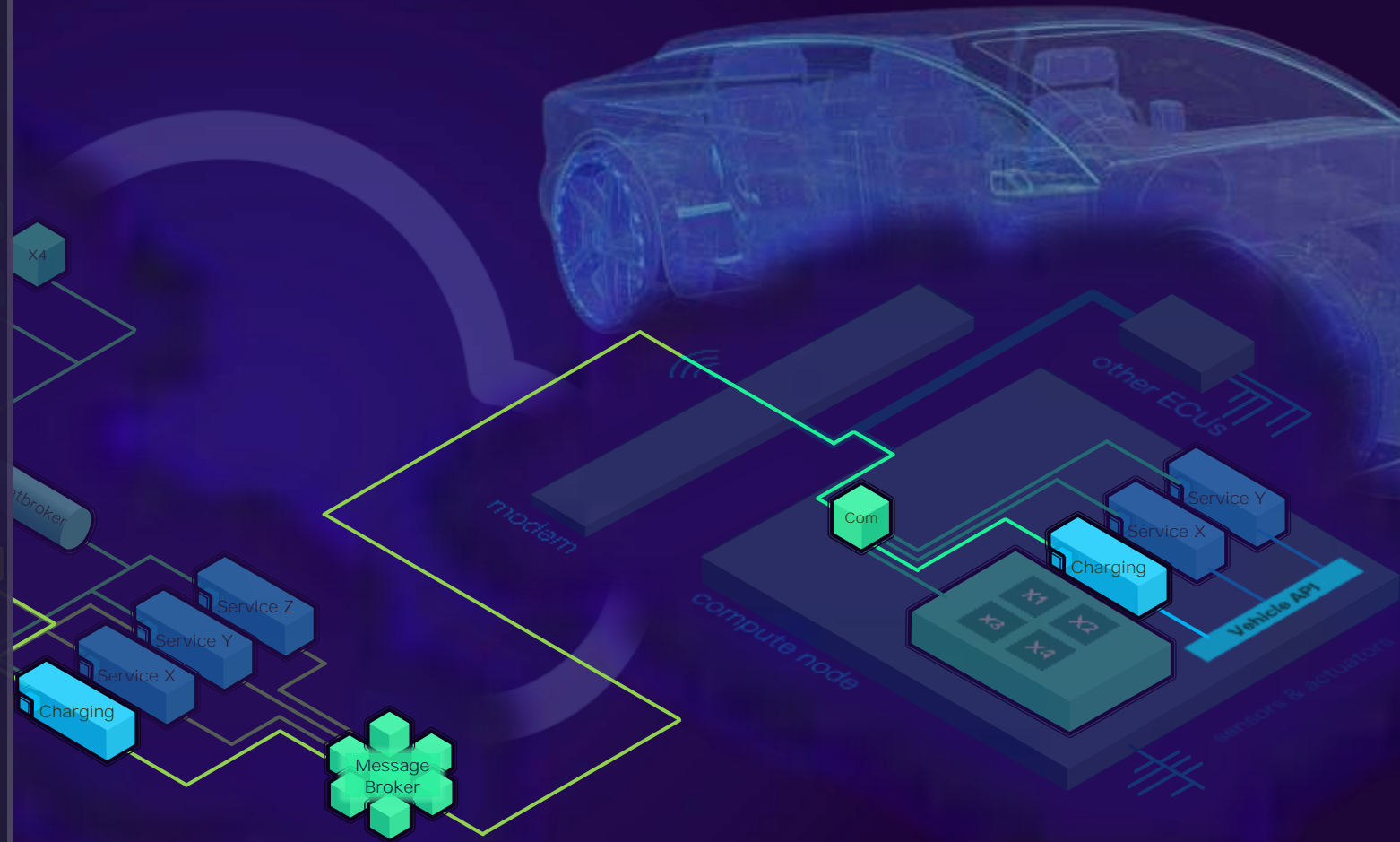
+

Unterschiedliche Namen für dasselbe System

- // Security nennt es „Battery Charge“
- // Legal nennt es „Batterieladen“
- // Entwickler nennen es „Charging“



Wir hatten kein
gemeinsames
Unternehmens-
Gedächtnis



Die Folge: Fragen blieben unbeantwortet

- // Wem gehört dieses System? Welche Kundenfunktionen hängen daran?
- // In welche Systeme sollte ich investieren, welche zusammenlegen oder abschalten?
- // Welche Clouddienste laufen in welchen Regionen?
- // Welche Kundenfunktionen sind bei einem Ausfall dieses Systems betroffen?



Vom Systeminventar zur Produktsicht

Jedes Asset erhält eine
eindeutige Identität

Nicht "Charging Service",
sondern `Application c112345` ist eindeutig

Adoption & Datenqualität

Das Tool war nie das
eigentliche Problem

Aber niemand wartet darauf:

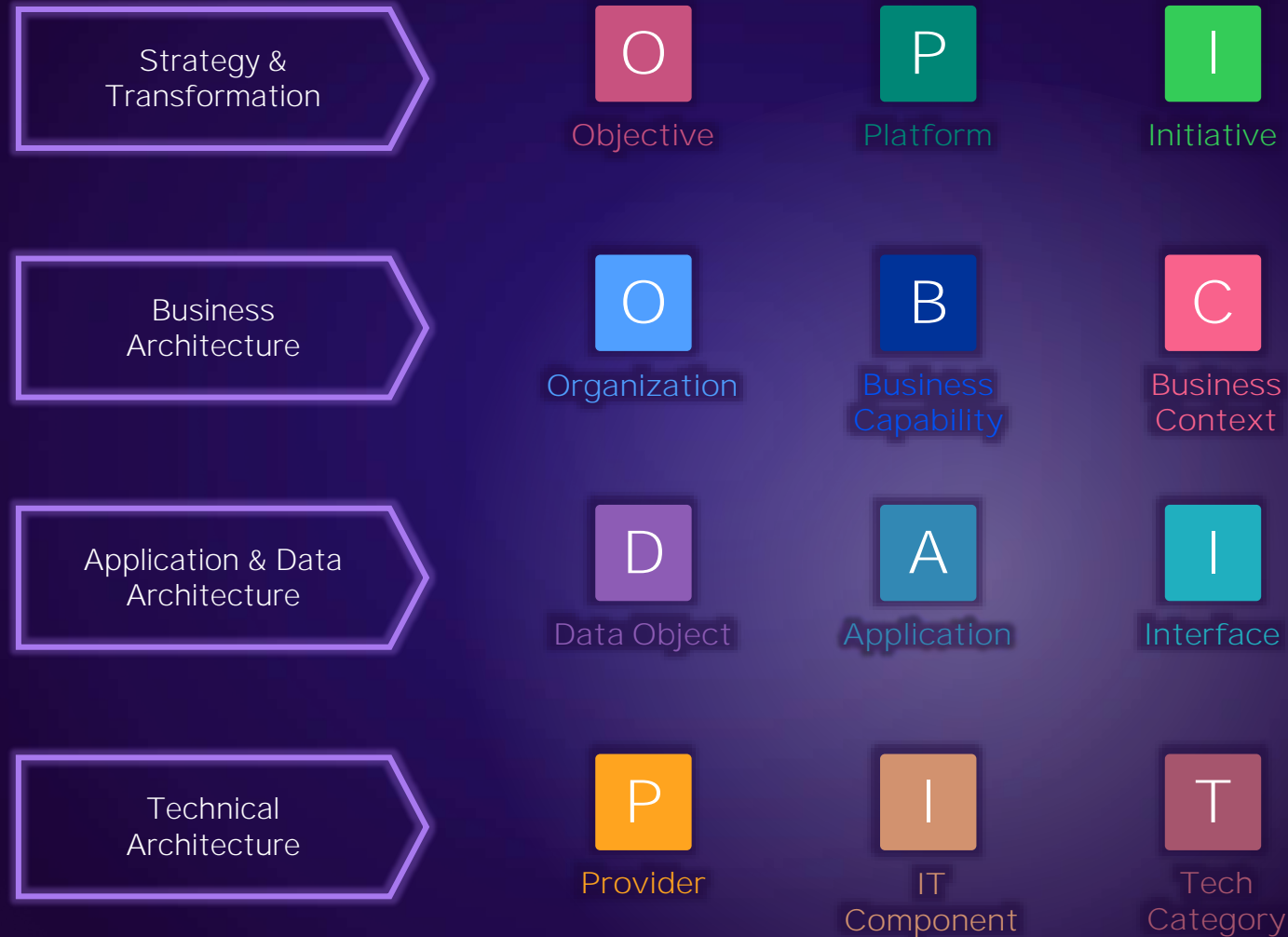
"Endlich darf ich Architektur dokumentieren."

Wie wir Akzeptanz geschaffen haben

Nicht durch: Governance, Pflichtfelder & Audits
sondern:

- Vorbefüllung des Modells
- Nutzen erzeugen
- Mehrwert sichtbar machen
- Dann erst Übergabe an Teams

Unser Produktzentrisches Metamodell



Unser Produktzentrisches Metamodell



Objective



Platform



Initiative



Organization



Business
Capability



Business
Context



Data Object



Application



Interface



Provider

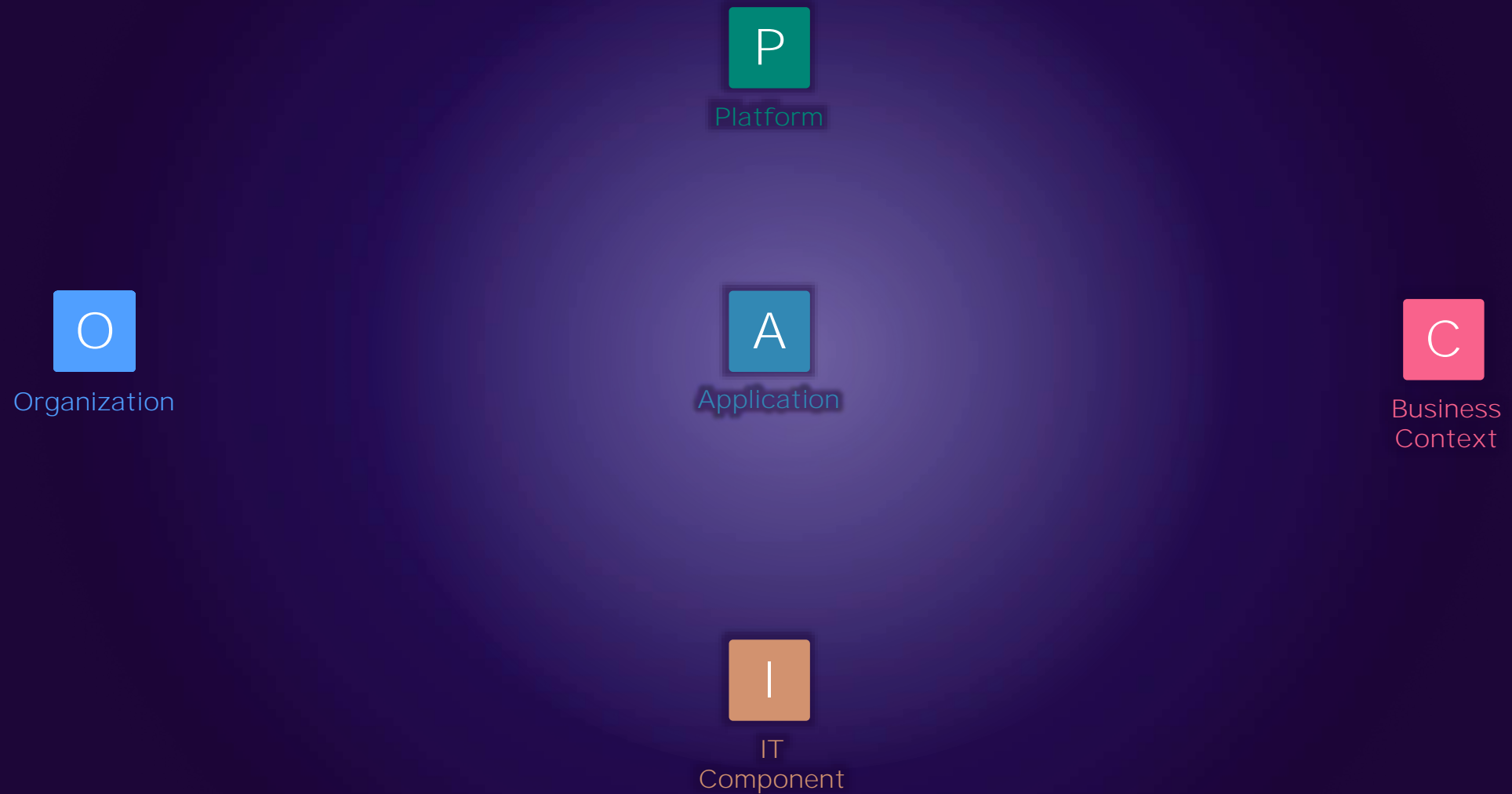


IT
Component

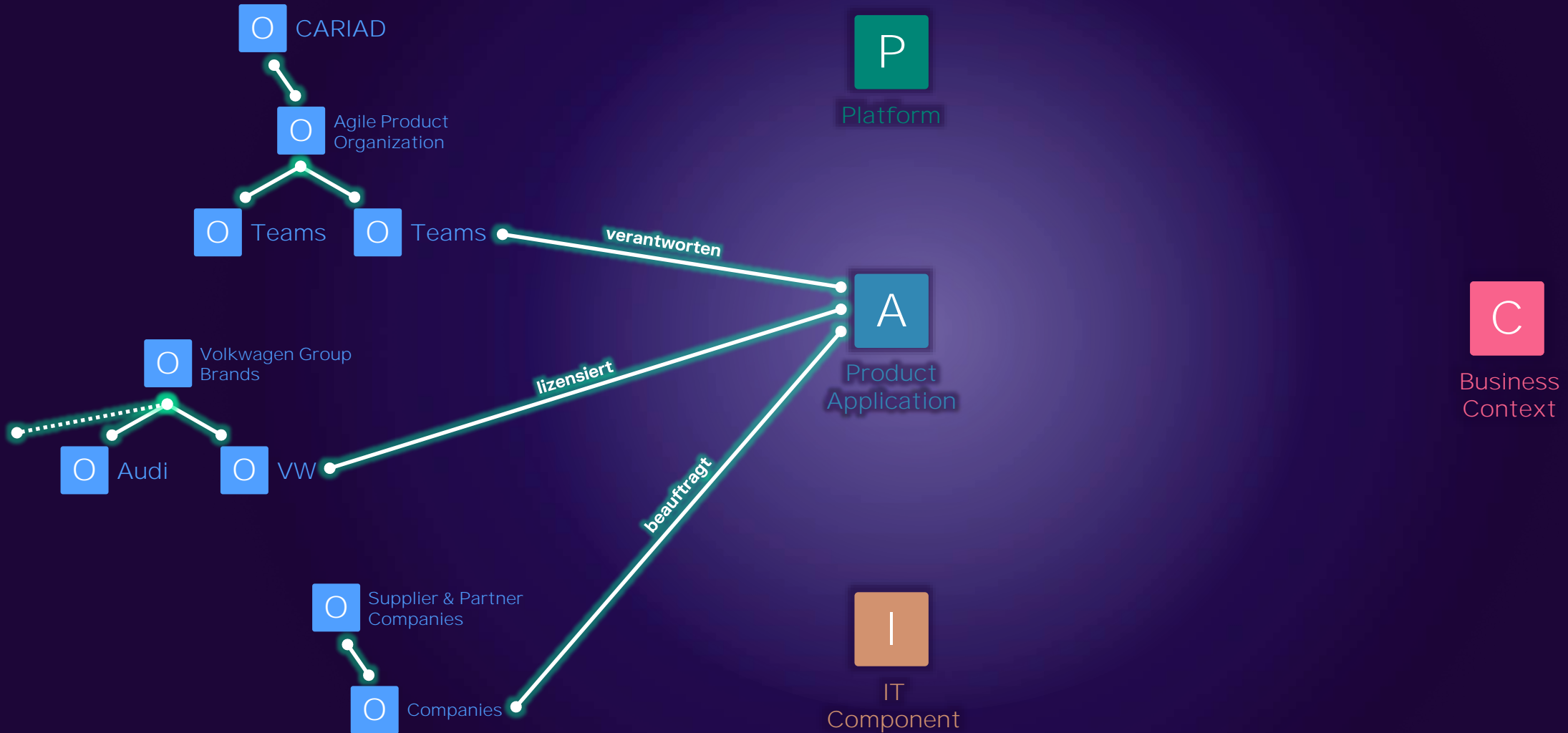


Tech
Category

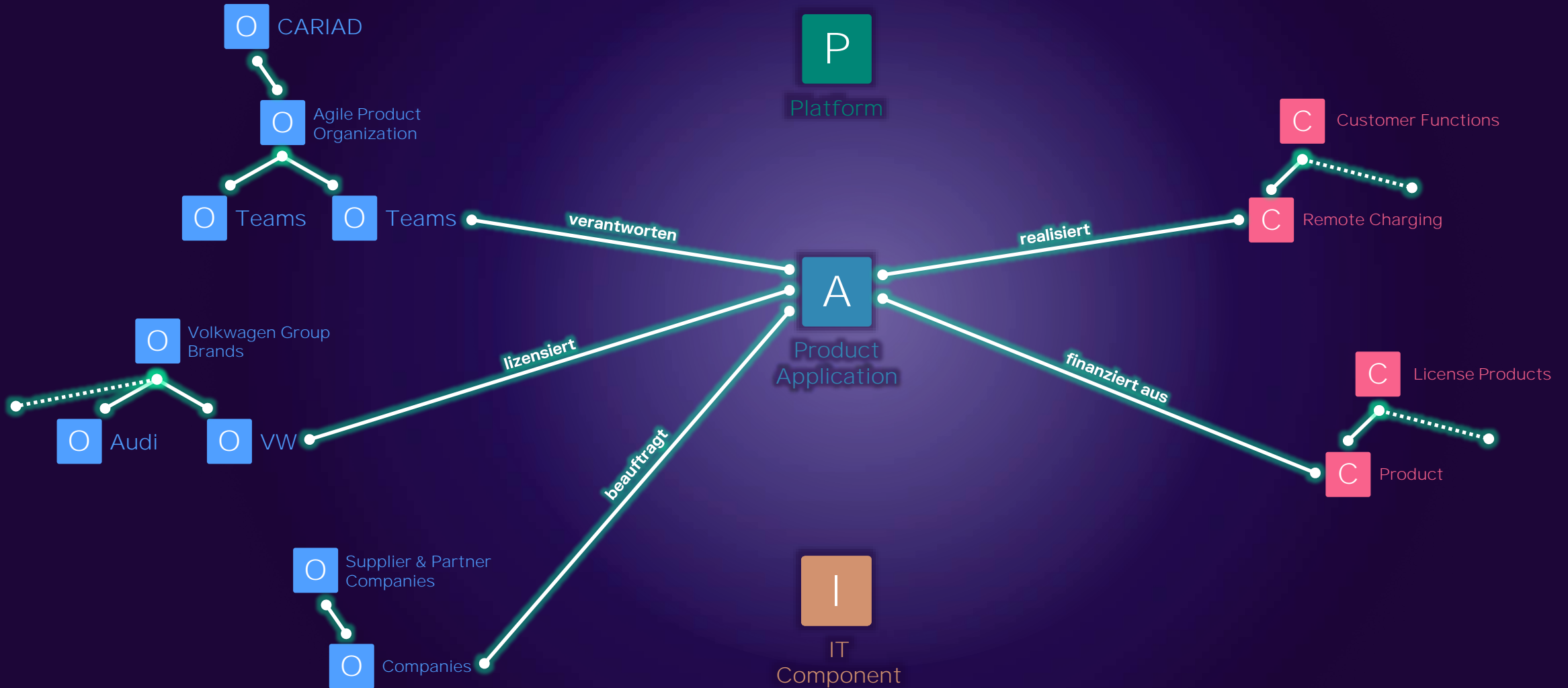
Unser Produktzentrisches Metamodell



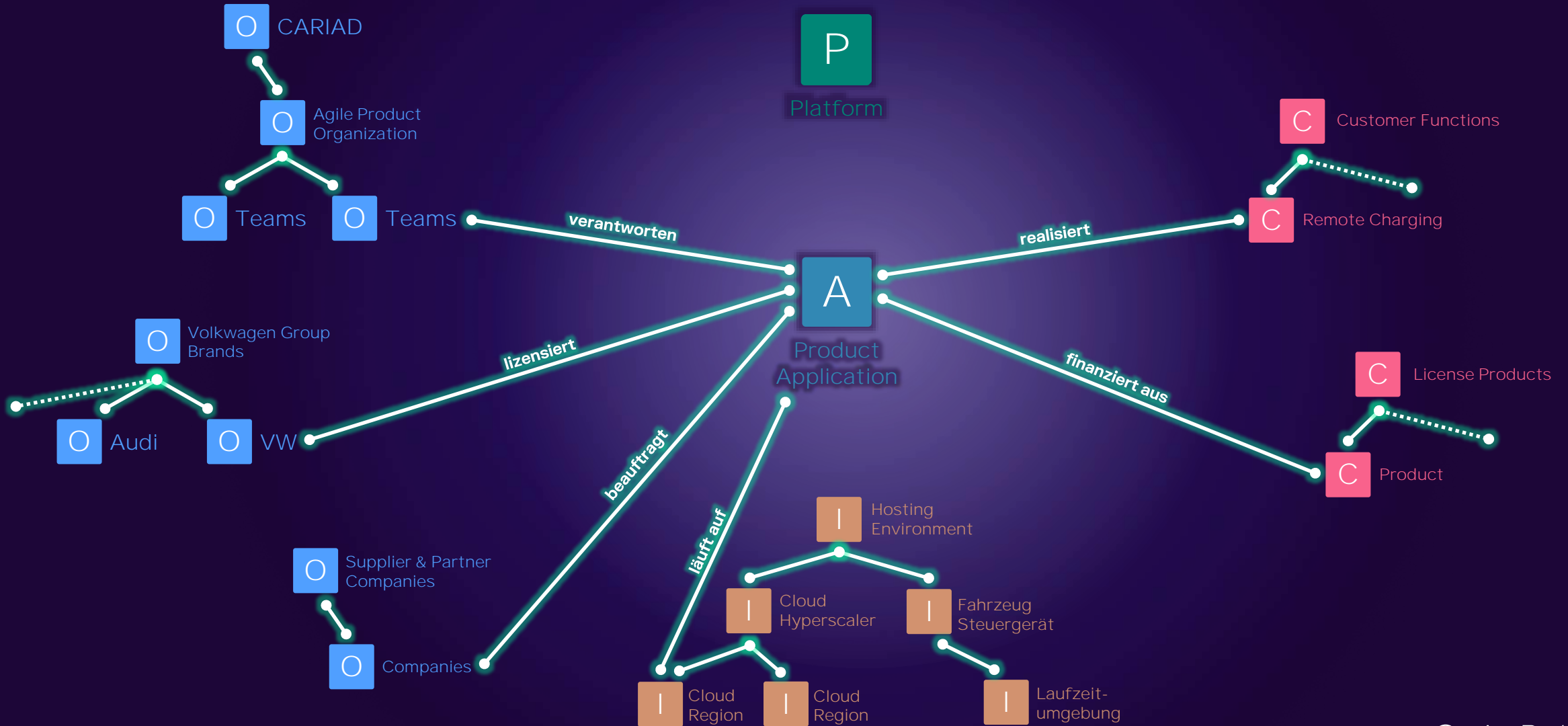
Unser Produktzentrisches Metamodell



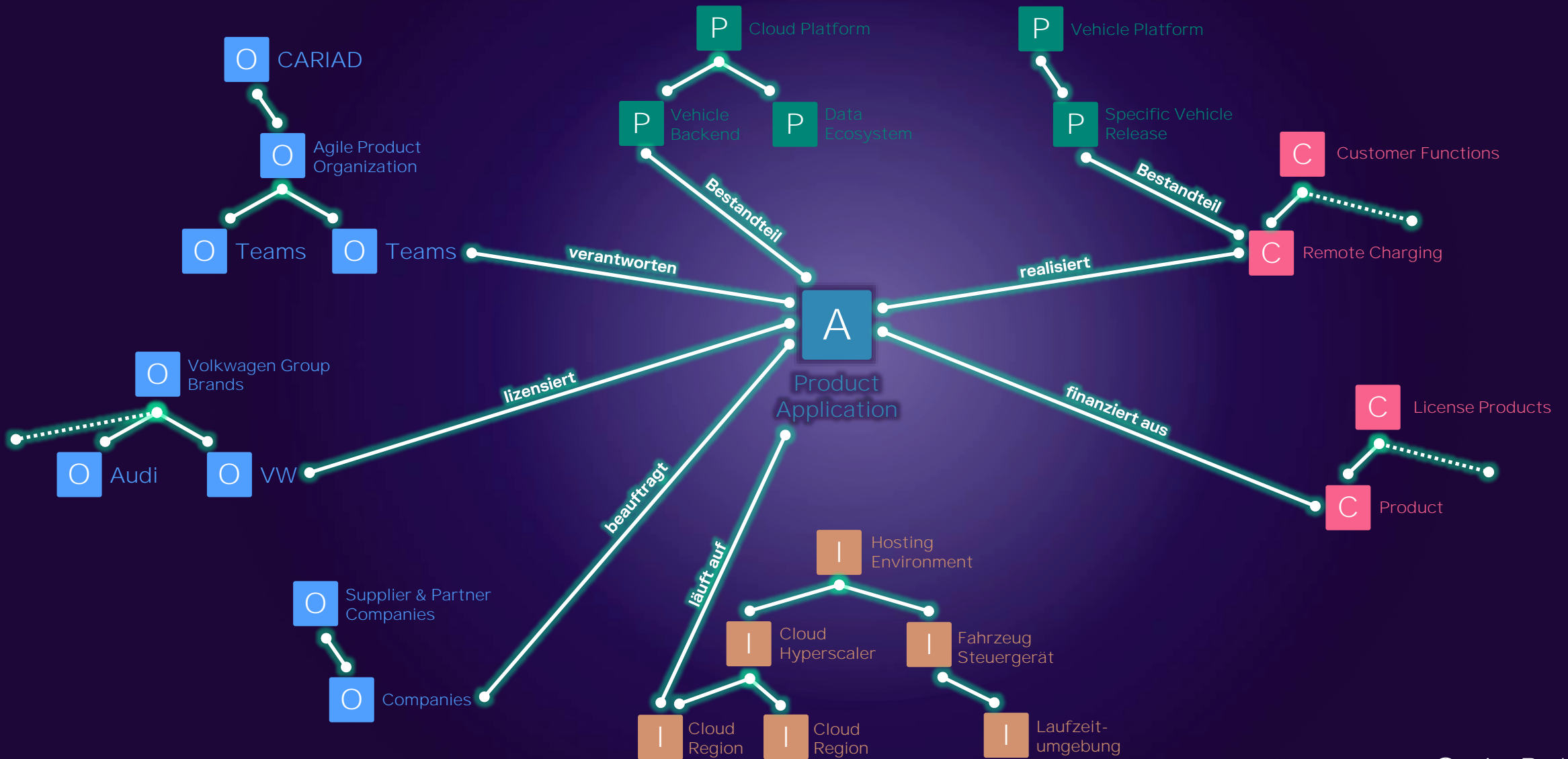
Unser Produktzentrisches Metamodell



Unser Produktzentrisches Metamodell



Unser Produktzentrisches Metamodell





OPEN SOURCE SCANS
Standardisierte SBOM Formate für Vulnerability Detection und Open Source License Texts



SOURCE CODE REPOSITORIES
Quellcode, CI/CD Pipelines



AI RELEVANZ
AI Modelle und Anwendungen
EU AI Act Risikoklassifizierung



LEGAL COMPLIANCE PRÜFUNGEN
Legal Assessments, Datenschutz, Exportkontrolle u.v.m.



SOFTWARE ARCHITEKTUR
Architekturdiagramme und Dokumentation




SECURITY ASSESSMENTS
Security & Safety, Pentests



WORK BREAKDOWN STRUCTURE
Jira Tickets, Arbeitspakete



API DOKUMENTATION
Developer Portal, Schnittstellenbeschreibung, technische Dokumentation,



FINANZDATEN
Entwicklungskosten
Betriebskosten
Lizenzentnahmen



DATENPUNKTE
Datenprodukte und Use Cases, Dateninhalte



WEITERE TOOLS & DATENQUELLEN
QA, Testergebnisse, Dokumente u.v.m.



ANFORDERUNGS-MANAGEMENT
Strukturierte Stakeholder Anforderungen, Funktional und Nicht-Funktional
Nachverfolgung und Traceability



OPERATIONS TOOLS
Monitoring
DevOps und FinOps Tooling
First-level Support



TEAMSTRUKTUR
Ablauforganisation, Release Trains, Teams

Was wir heute beantworten können



Welche Kundenfunktionen hängen von welchen Applikationen ab?



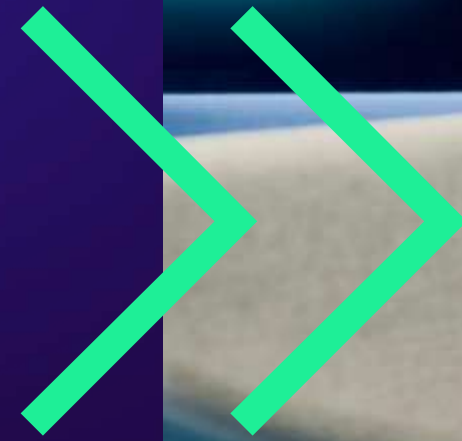
Welche Abhängigkeit habe ich zwischen den Teams, welche zwischen den Systemen?



Welche Produkte verarbeiten personenbezogene Daten? Welche nutzen KI?



Welche Kosten verursacht ein Produkt?



Unsere nächste Ausbaustufe

- // Weiterer Ausbau automatisierter Integrationen
- // KI-gestützte Architekturdokumentation
- // Analysen weiter anreichern und tiefere Insights für strategische Entscheidungen gewinnen
- // Synchronisation aus Produktivsystemen wie Logging, Monitoring und CI/CD

und vieles weiteres mehr



Enterprise Architecture ist nicht
die Dokumentation von Systemen.

Enterprise Architecture schafft ein
gemeinsames Verständnis des Produkts.

Und **gemeinsame Entscheidungen** beginnen
immer mit einem gemeinsamen Verständnis.

Vielen Dank!



 RETHINK
EAM

CARIAD