How Snapdragon® Chassis Agents Bring Agentic AI to the AI-Defined Vehicle

The Next Step in Automotive Intelligence
with Qualcomm Technologies

By Girish Shirasat

Senior Director, Engineering

Qualcomm Technologies

Table of Contents

Introduction

The software-defined vehicle (SDV) is becoming a given in the automotive industry, and with Qualcomm Technologies as a leading technology provider. The next paradigm shift is toward the AI-defined vehicle.

The AI-defined vehicle (AIDV) is one whose functionality and user experience are primarily determined by AI models, autonomous agents and data-driven processes. This means that the vehicle's behavior, features and interactions are dynamically shaped by AI, allowing for continuous improvement and adaptation. The AIDV can take inputs from multiple contexts for autonomous functions as varied as monitoring vehicle health, analyzing road conditions, intervening in emergencies, recognizing passengers and launching personal assistants. AIDV software combines advanced connectivity with software-defined and agentic AI architectures.

To give original equipment manufacturers (OEMs) the components needed for the AIDV, Qualcomm Technologies is offering Snapdragon® Chassis Agents. The agents build on the widely successful Snapdragon Digital Chassis product portfolio of system on chips (SoC). Snapdragon Digital Chassis provides the mixed-criticality, heterogeneous compute needed to execute edge AI in the vehicle, along with the Snapdragon Ride™ ADAS stack and Connected Services infrastructure.

By combining agentic frameworks and on-edge AI inferencing, Snapdragon Chassis Agents enable independent, goal-driven software entities that orchestrate and personalize vehicle functions in real time. They implement the concept of compound AI as real-world, agentic AI systems, enabling the vehicle to see layers of detail through its sensors and translate that detail into human- and machine-consumable contexts. Everything the car perceives in its environment and in the cabin, combined with the passenger's digital life, is used to improve the overall in-vehicle experience.

This paper describes Snapdragon Chassis Agents: their agentic architecture and their relationship to the Snapdragon Digital Chassis. OEMs will see how they can use the agents to enhance their own agentic AI with in-vehicle data across domains. As vehicles become increasingly defined by AI, Snapdragon Chassis Agents have the potential to help deliver greater safety, intelligence and engagement.

AI technologies and their relationships

First, consider the relationships among the different types of AI at work in automobiles, as depicted in Figure 1:

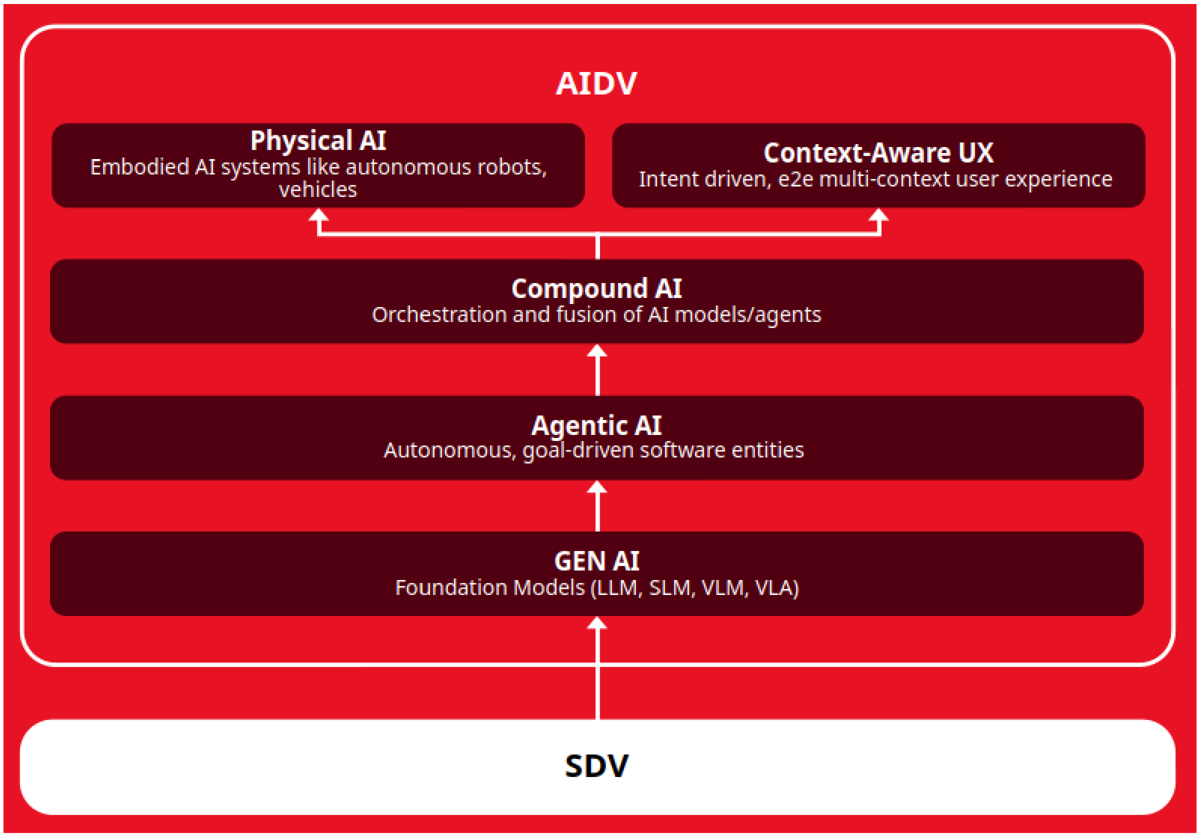

Figure 1: AI technologies and relationships

Tracing the evolution of those technologies from the bottom upward,
here are their definitions:

- Software-defined vehicle (SDV) – A vehicle that relies on software to define and update its functions and user experience throughout its lifecycle. Unlike traditional vehicles with fixed features, SDVs can receive updates and new capabilities via software, so that they remain current and adaptable.
- Generative AI (Gen AI) – AI models that can generate new content—such as text, images, or audio—by learning patterns from large datasets.

- Agentic AI – Independent, goal-driven software entities capable of planning, reasoning and acting to achieve specific objectives. Unlike traditional rule-based systems, agentic AI can adapt strategies based on changing environments and goals, making it suitable for complex, real-world tasks such as those encountered in vehicles.
- Compound AI – A practical instantiation formed by Gen AI and agentic AI together, in which multiple specialized AI models or agents are orchestrated to reach a particular goal. The approach takes advantage of the relative strengths of each component to perform complex tasks that would be difficult for a single model, resulting in greater robustness and flexibility.
- Physical AI **–** AI systems embedded within physical entities such as robots, machines, vehicles and drones. The systems can perceive their environment, reason about it and take actions in the real world, bridging the gap between digital intelligence and physical operation.
- Context-Aware UX – Naturalistic, interactive vehicle interface that understands who the user is, what they're doing and what matters now. It adapts interactions dynamically based on the environment within and outside of the car.
- AI-defined vehicle (AIDV) – A vehicle whose functionality and user experience are primarily determined by AI models, agentic AI and data-driven processes. The vehicle's behavior, features and interactions are dynamically shaped by AI, allowing for continuous improvement and adaptation.

Thus, the AIDV builds upon the foundation of the SDV. Note also that Gen AI, agentic AI, compound AI and physical AI all depend on and incorporate multiple AI models, including:

- Large language models (LLM) – Powering natural language interfaces and in-vehicle assistants.
- Small language models (SLM) – Optimized for on-device inference, enabling real- time interactions with minimal latency.
- Video language models (VLM) – Multimodal models capable of interpreting and generating information from both video and language data. They enable the system to understand complex visual scenes in conjunction with textual or spoken instructions.
- Video language action models (VLA) – Models that extend VLMs by adding action recognition and prediction. That allows the system to not only interpret but also anticipate and respond to events in the vehicle's environment. The outputs of VLA and VLM, along with the reasoning capabilities of compound AI, are integrated into physical AI.

Compound/Agentic AI – Examples

Compound AI, then, refers to the orchestration of multiple specialized AI agents and models to deliver intelligent, context-aware services that enhance safety and user experience. Consider these examples:

- Passenger identification and personalized interaction – The vehicle can use agents to recognize and authenticate passengers using visual and auditory cues. Through on-device Gen AI and text-to-speech, the system can greet each seat occupant by name and tailor its responses individually.
- Integration of calendar and appointment data – Agents can act as personal assistants, monitoring upcoming events and suggesting actions such as navigation to scheduled appointments. When a calendar event is detected, the system offers reminders and route planning to the driver.
- Adaptive comfort management – In response to bumpy road conditions and driver commands, the system can take advantage of personal assistant agents, advanced driver assistance system (ADAS) modules and on-device SLMs. Automatically switching to comfort mode, it can improve ride quality by adjusting factors like vehicle speed.
- Scene analysis and contextual assistance – Using multimodal perception and memory extraction, a suite of cooperating agents can extract information from external objects, such as details of a passing truck or a roadside hazard. It can then assist with placing a phone call or setting reminders.
- Vehicle health monitoring – Continuously tracking indicators like tire pressure and fluid levels, the system can detect anomalies, engage ADAS, adjust cruise control and provide the driver with recommended actions. It can use on-device language models to consult the vehicle manual and relay procedures to the driver.
- Coordination of rapid intervention – Orchestrating cockpit, wearables, telematics and ADAS agents to safeguard occupants, the system can watch for signs of driver distress, like drowsiness or abnormal vital signs. It can then initiate measures such as vibrating the seat, activating hazard lights, making emergency calls and transmitting driver information to medical services.

System architecture for automotive agentic AI

Given the above-described definitions and relationships among types of AI, what needs to be in place for OEMs to build the AIDV that delivers those examples? Which architectural elements does agentic AI in automotive require?

Figure 2 illustrates a hybrid agent architecture for automotive systems. The architecture spans both in-vehicle and cloud-based intelligence for advanced, distributed decision making and orchestration across domains.

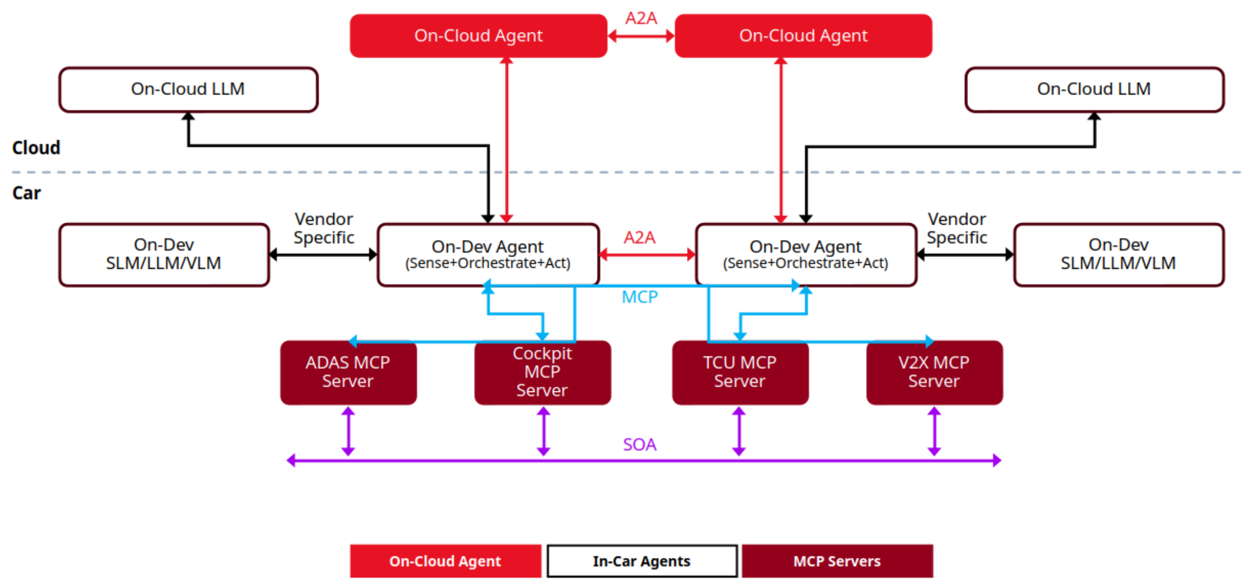

Figure 2: Architecture for automotive agentic AI

In-car layer

Inside the vehicle, on-device agents (in white) are deployed, each equipped with specialized models such as SLMs, LLM and VLMs. The agents detect and act on real-time data from the vehicle, using vendor-specific integrations to optimize performance and delivery of automotive features.

The on-device agents interface directly with domain-specific Model Context Protocol (MCP) servers (in dark red), including these:

- ADAS MCP Server for advanced driver assistance
- Cockpit MCP Server for infotainment and user experience
- Telematics control unit (TCU) MCP Server for telematics and connectivity
- V2X MCP Server for vehicle-to-everything communications

The MCP servers provide the necessary abstraction and control points for modular access to vehicle functions.

Cloud layer

The architecture extends to the cloud, where on-cloud agents and Gen AI models reside (in light red). The agents can communicate with their in-vehicle counterparts via agent-to-agent (A2A) protocols, enabling distributed reasoning, remote orchestration and feature updates.

Vendor-specific pathways allow for tailored integration between cloud and on-device models, for proprietary enhancements and differentiated services. The cloud agents can take advantage of powerful LLMs for complex reasoning, data enrichment and cross-vehicle coordination, while maintaining efficient communication with the vehicle's on-device agents.

Example: Accessing the rear camera view

Consider the simple use case of accessing the rear camera view. Figure 3 illustrates the difference between the legacy approach of a physical button press using a service-oriented architecture (SOA) workflow and a natural language prompt processed by an agentic AI workflow in an automotive context.

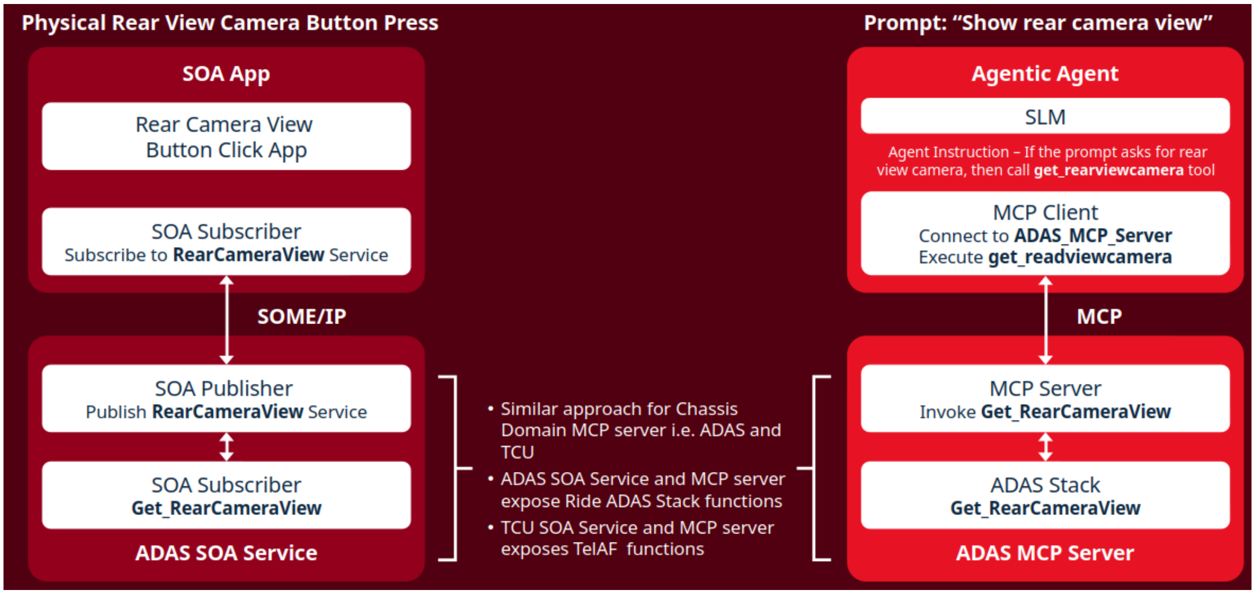

Figure 3: Accessing the rear camera view

SOA workflow

On the left, the user presses a button, which triggers an SOA app for accessing the physical rear camera view. The app is a subscriber to the RearCameraView service. Communication between the app and the underlying ADAS stack is facilitated by the SOME/IP protocol, a standard for automotive service-oriented messaging. The SOA Publisher exposes the RearCameraView service, implemented in the ADAS stack through the Get_RearCameraView function.

The ADAS SOA Service thus enables modular, event-driven access to camera functions. This approach is mirrored for other MCP servers in the chassis domain, such as ADAS to expose Snapdragon Ride ADAS Stack functions and TCU to expose Telematics Application Framework (TelAF) functions.

Agentic AI workflow

On the right, the user utters a natural language prompt such as "Show me rear camera view." The agent, powered by an SLM, interprets the instruction and determines that the get_rearviewcamera tool should be invoked. The agent communicates with the MCP Client, which connects to the ADAS MCP Server and executes the get_rearviewcamera command. The MCP Server then invokes the Get_RearCameraView function in the ADAS Stack, mirroring the underlying service call of the SOA workflow.

Difference in approach

Both approaches ultimately access the same ADAS Stack function but differ in their initiation and orchestration.

The agentic AI workflow takes advantage of natural language understanding and tool invocation through an agentic planning layer. That enables flexible methods of interaction between user and vehicle, whether through traditional UI controls or conversational AI interfaces. The SOA workflow, on the other hand, is event-driven and relies on explicit service subscriptions and publications. Both are essential to the AIDV: SOA enables the current, industry-standard way of building in-car distributed applications, which can then be exposed to agentic AI infrastructure through MCP servers.

The underlying services of the automotive agentic AI architecture are modular and reusable, regardless of the interaction modality. The services are designed with safety in mind and orchestrated to access critical vehicle features.

The software stack (generic) for the AI-defined vehicle

Next, Figure 4 depicts a generic stack of the software required to enable the automotive agentic AI architecture.

The software consists of two layers: AIDV infrastructure for advanced, AI-driven vehicle capabilities, and SDV foundational infrastructure for the vehicle's essential software and hardware backbone.

Together, they run atop a central compute SoC designed for mixed-criticality automotive applications.

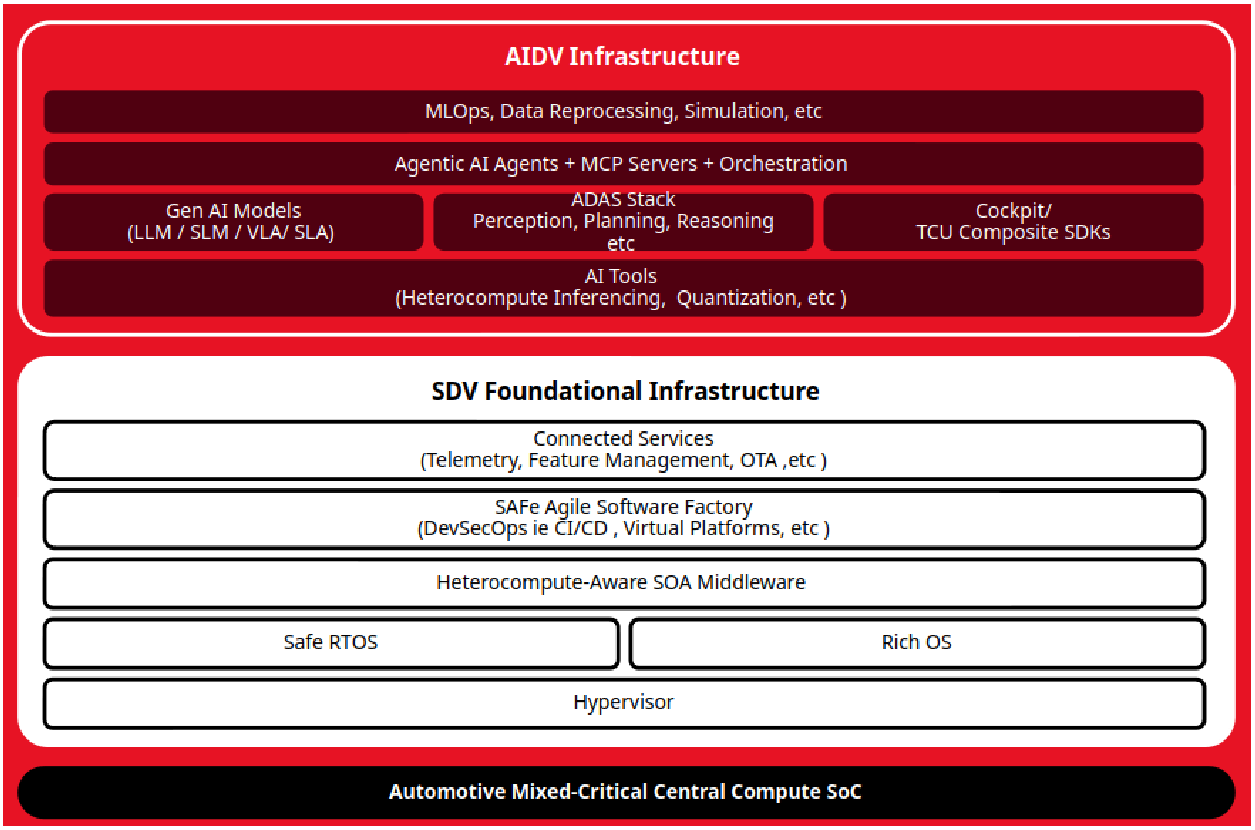

Figure 4: Software stack (generic) for AIDV

AIDV infrastructure

AIDV infrastructure incorporates machine learning operations (MLOps), data reprocessing and simulation for continuous model enrichment and rapid iteration of AI models. Agentic AI, together with MCP servers and orchestration layers, uses vehicle data to enable independent, goal-driven behaviors across the system.

The infrastructure supports generative AI models including LLMs, SLMs, VLMs and specialized VLA models. The models are optimized for the real-time, on-device inferencing needed in automotive applications. The ADAS stack provides perception, planning and reasoning functions. Composite software development kits (SDKs) for cockpit and TCU facilitate rapid development and integration of user-facing features. AI tools for heterogeneous compute inferencing and quantization enable efficient utilization of hardware accelerators and support for mixed-criticality workloads.

SDV foundational infrastructure

SDV foundational infrastructure includes connected services such as telemetry, feature management and over-the-air updates, for continuous delivery and remote management of vehicle features. The SAFe Agile Software Factory supports DevSecOps practices, CI/CD pipelines and virtual platforms for agile feature development and deployment. Hardware-aware, safe SOA middleware acts as the data fabric, orchestrating communication and computation across diverse vehicle domains.

The layer hosts both safe real-time operating systems (RTOS) and rich operating systems, enabling robust support for safety-critical ADAS and cockpit applications. A hypervisor provides isolation and security for mixed-criticality workloads, allowing multiple software stacks to co-exist safely.

Advantages of the architecture

The architecture emphasizes OEM-controlled feature management, agile software factories for rapid delivery and hardware-aware middleware for efficient data handling. On-device optimized AI stacks for LLM, ADAS and cockpit domains enable low-latency, privacy-preserving intelligence. The system is designed for continuous model enrichment, using vehicle data to enhance agentic AI capabilities and support evolving user experiences. Safety and security are prioritized through the use of safe operating systems, hypervisors and automotive-grade SoCs with integrated AI accelerators.

Overall, this architecture enables a modular, scalable, security-rich foundation for software-defined and AI-driven vehicles. It is designed for rapid innovation, personalized user experiences and robust safety throughout the vehicle.

How Snapdragon Digital Chassis enables the AIDV software stack

How does Qualcomm Technologies deliver on that software stack? How does the Snapdragon Digital Chassis let OEMs take advantage of agentic AI and build the AIDV?

The Snapdragon Digital Chassis-enabled AIDV stack is a multi-layered automotive platform designed to deliver scalable, safety-focused, intelligent vehicle experiences. Building on the same two layers described above – AIDV infrastructure and SDV foundational infrastructure – the stack integrates heterogeneous compute, advanced middleware and AI-driven services.

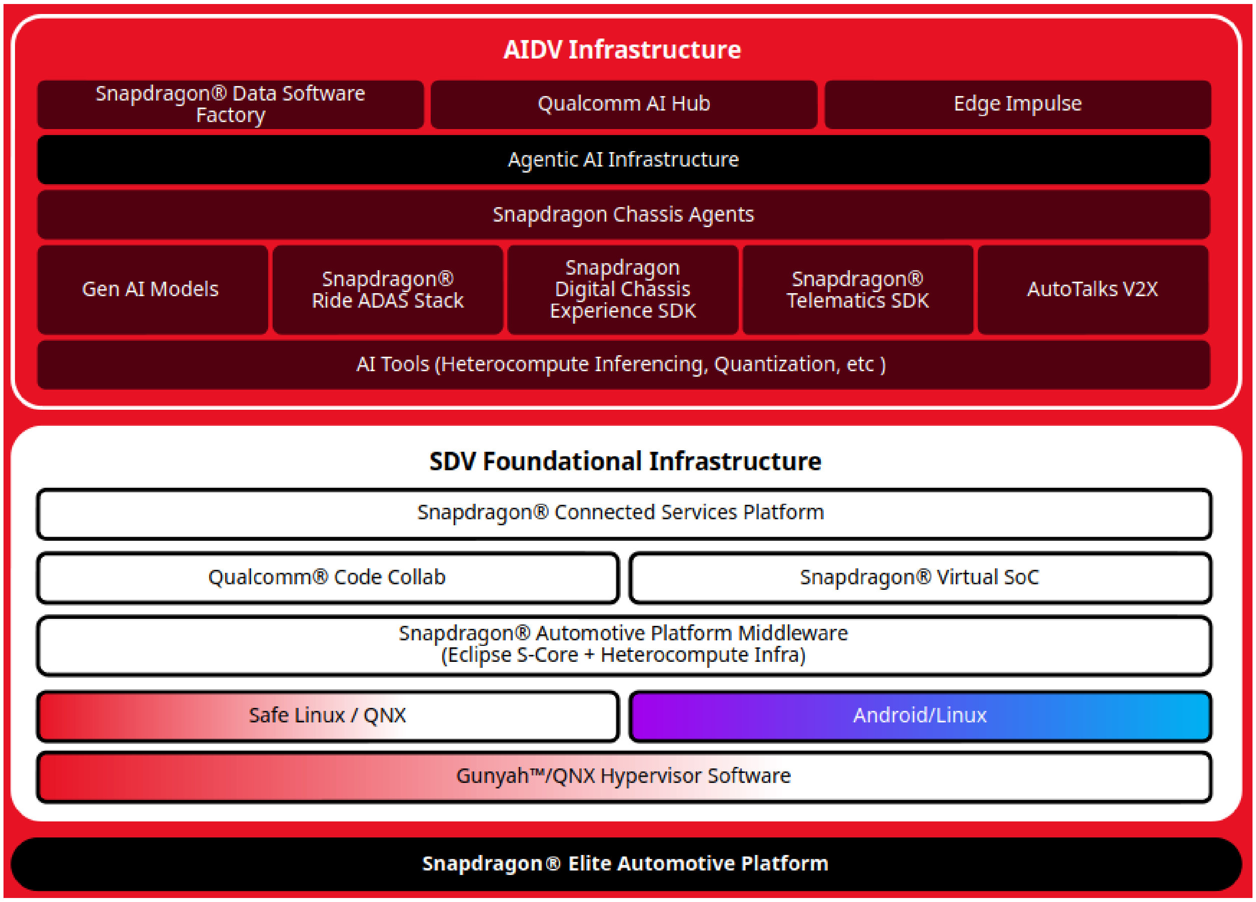

Figure 5: Software stack for AIDV, as enabled by Snapdragon Digital Chassis

AIDV infrastructure

The Snapdragon Data Software Factory supports ADAS model data operations and MLOps for continuous model enrichment and simulation. The Qualcomm AI Hub allows for the execution and testing of AI models on automotive devices in the cloud, with both on-device and cloud-based workflows. Edge Impulse provides time-series model MLOps, for rapid development and deployment of edge AI models.

The agentic AI infrastructure includes Snapdragon Chassis Agents: software agents and MCP servers for domains such as ADAS, cockpit and telematics. They orchestrate perception, planning and actuation throughout the vehicle (see “Snapdragon Chassis Agents” below).

The Snapdragon Ride ADAS Stack provides end-to-end, model-based perception and planning for advanced driver assistance features. The Snapdragon Digital Chassis Experience SDK offers a composite, context-aware cockpit service development kit, while the Snapdragon Telematics SDK provides a telephonic SDK for TCU SoC integration. AutoTalks SDKs are included for developing host-side V2X applications on Qualcomm® V2X SoCs, enabling vehicle-to-everything communication and safety features. The infrastructure supports a variety of generative AI models (LLM, SLM, VLA/VLM), optimized for on-device inferencing and quantization to meet real-time automotive requirements.

SDV foundational infrastructure

The Snapdragon Connected Services Platform manages telematics, over-the-air (OTA) updates, feature management and other connected services. Qualcomm Code Collab is a SAFe Agile-based software factory that enables DevSecOps practices and continuous integration/delivery. The Snapdragon Virtual SoC allows for cloud-based virtual SoC development, enabling environmental parity between cloud and edge deployments.

The Platform board support package (BSP) enables mixed criticality by providing platform software that comprises hypervisor, OS and platform libraries.

The foundational layer hosts both safe Linux/QNX and Android/Linux operating systems, ensuring robust support for safety-critical ADAS and cockpit applications. Gunyah/QNX hypervisors provide isolation and security for mixed-criticality workloads, enabling safe co-existence of multiple software stacks.

Snapdragon® Automotive Platform Middleware

Within the SDV foundational infrastructure, Snapdragon Automotive Platform Middleware (APM) acts as the primary data fabric across the Snapdragon Digital Chassis.

In multiple layers (see Figure 6), the middleware is designed to optimize and orchestrate Qualcomm® hardware resources in automotive environments. It provides a foundation for building AI-driven vehicle systems by abstracting hardware complexity and enabling efficient compute management, service orchestration and communication.

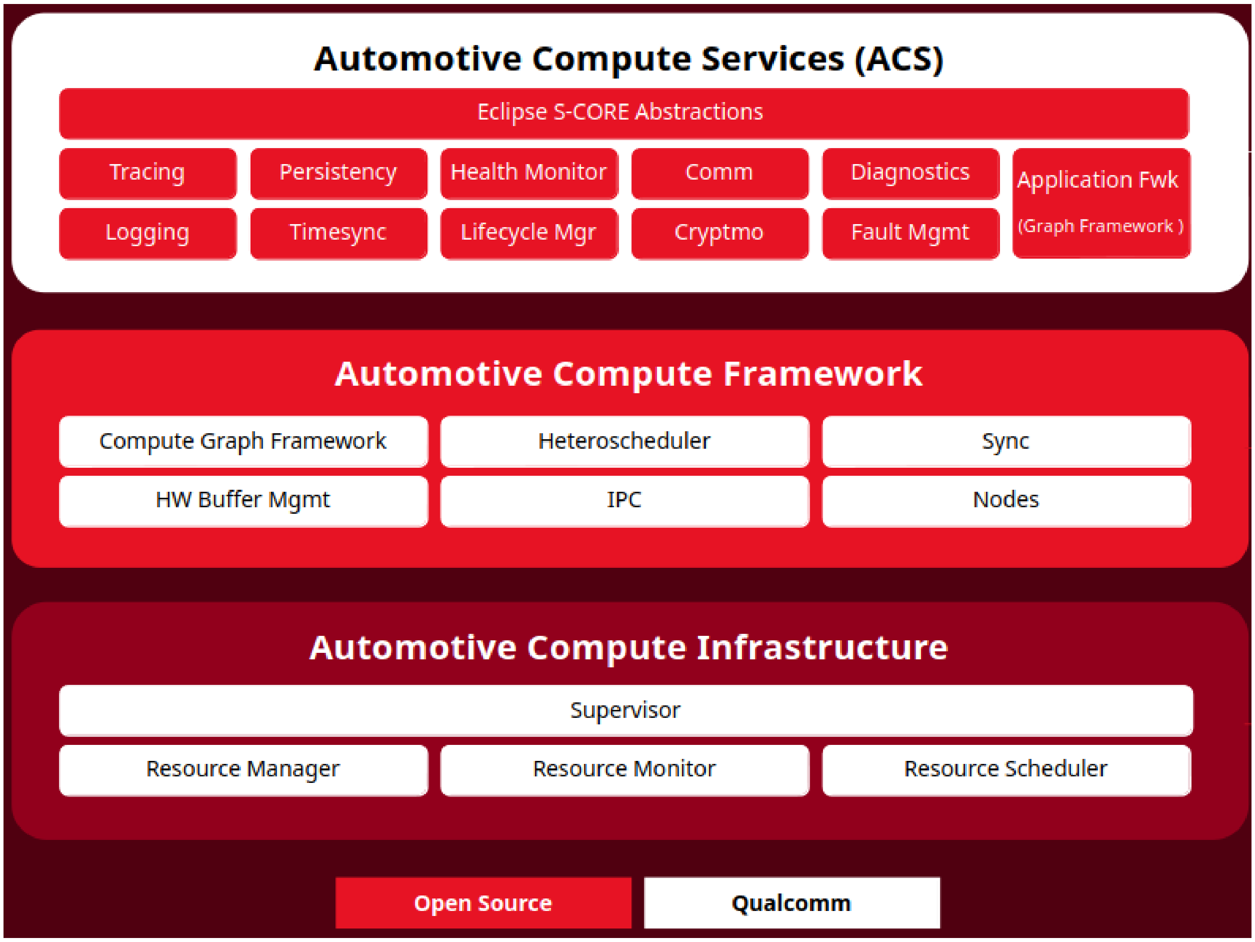

Figure 6: Snapdragon Digital Chassis Automotive Platform Middleware

Automotive Compute Services (ACS)

These middleware services include tracing, logging, persistence, health monitoring, communication, diagnostics, timesync, lifecycle management, cryptography and fault management. They are built on Eclipse S-CORE abstractions, enabling compatibility and extensibility across diverse hardware and software environments. The application framework, including the graph framework, supports modular application development and orchestration.

Automotive Compute Framework (ACF)

The framework enables optimal graph execution across heterogeneous hardware. It includes a compute graph framework, a heteroscheduler for dynamic resource allocation, synchronization mechanisms, hardware buffer management and inter-process communication (IPC).

Automotive Compute Infrastructure (ACI)

The infrastructure provides the supervisor for the resource manager, resource monitor and resource scheduler (“Resource Trifecta”) for hardware in a flex type of multi-domain consolidated system.

Snapdragon Chassis Agents

Snapdragon Chassis Agents take compound AI from the conceptual level to the reality of a product. They allow OEMs to process the analog environment for multiple contexts and create a richer, more-reasoned result.

Based on the System architecture for automotive agentic AI (see above), Snapdragon Chassis Agents capture data and perform actions in user-facing agentic applications. As shown in Figure 7, they work across in-vehicle domains such as ADAS, cockpit, telematics and vehicle-to-everything (V2X) communications.

Snapdragon Chassis Agents represent a pivotal advancement in the orchestration of intelligent, vehicle-centric software systems.

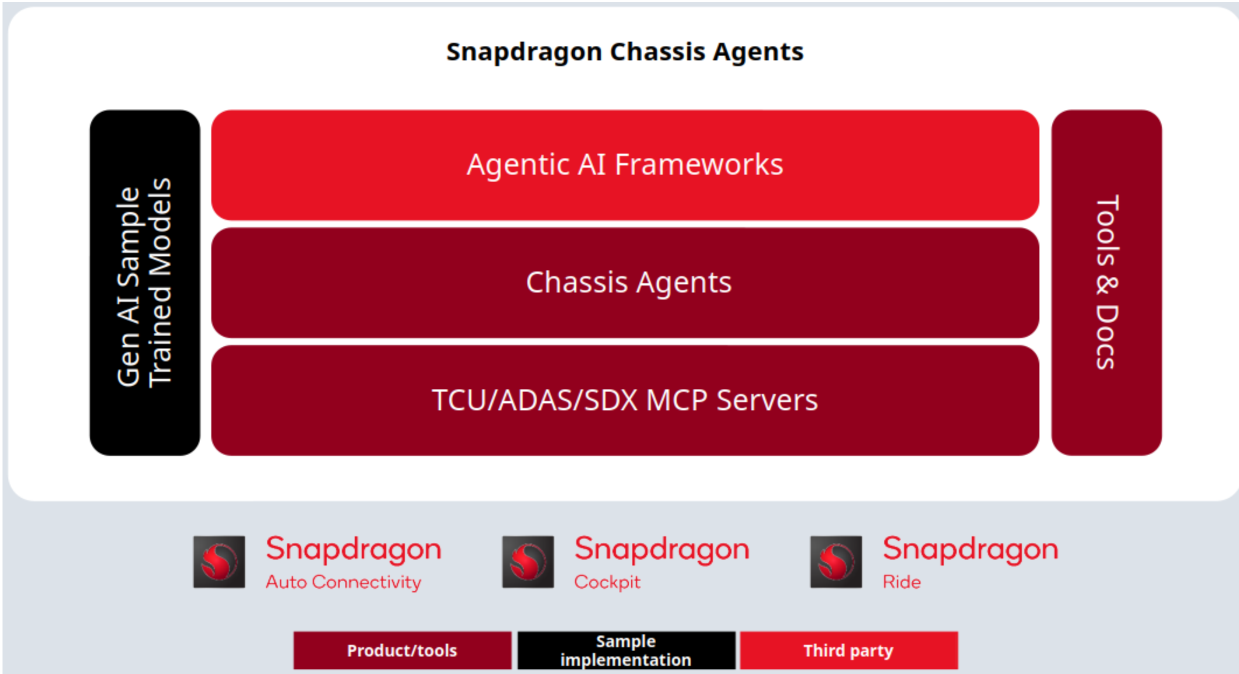

Figure 7: Snapdragon Chassis Agents

Designed to be framework-agnostic, the Snapdragon Chassis Agents product includes documentation, development tools, sample implementations and third-party integrations. It encompasses a collection of agents atop MCP servers tailored for domains such as TCU, ADAS and in-vehicle infotainment (IVI). It also supports the deployment of generative AI models and agentic frameworks such as Google's Automotive AI Agent.

This architecture is designed to be extensible, allowing for the integration of both Qualcomm Technologies products and third-party solutions. By providing a unified, agent-driven approach to vehicle intelligence, Snapdragon Chassis Agents accelerate the development and deployment of advanced automotive features, enhance safety and personalization and lay the groundwork for AIDVs.

Figure 8 shows a system architecture in which the Snapdragon Chassis Agent suite manages automotive functions across multiple domains, including telematic, ADAS, cockpit and cluster/driver monitoring system (DMS).

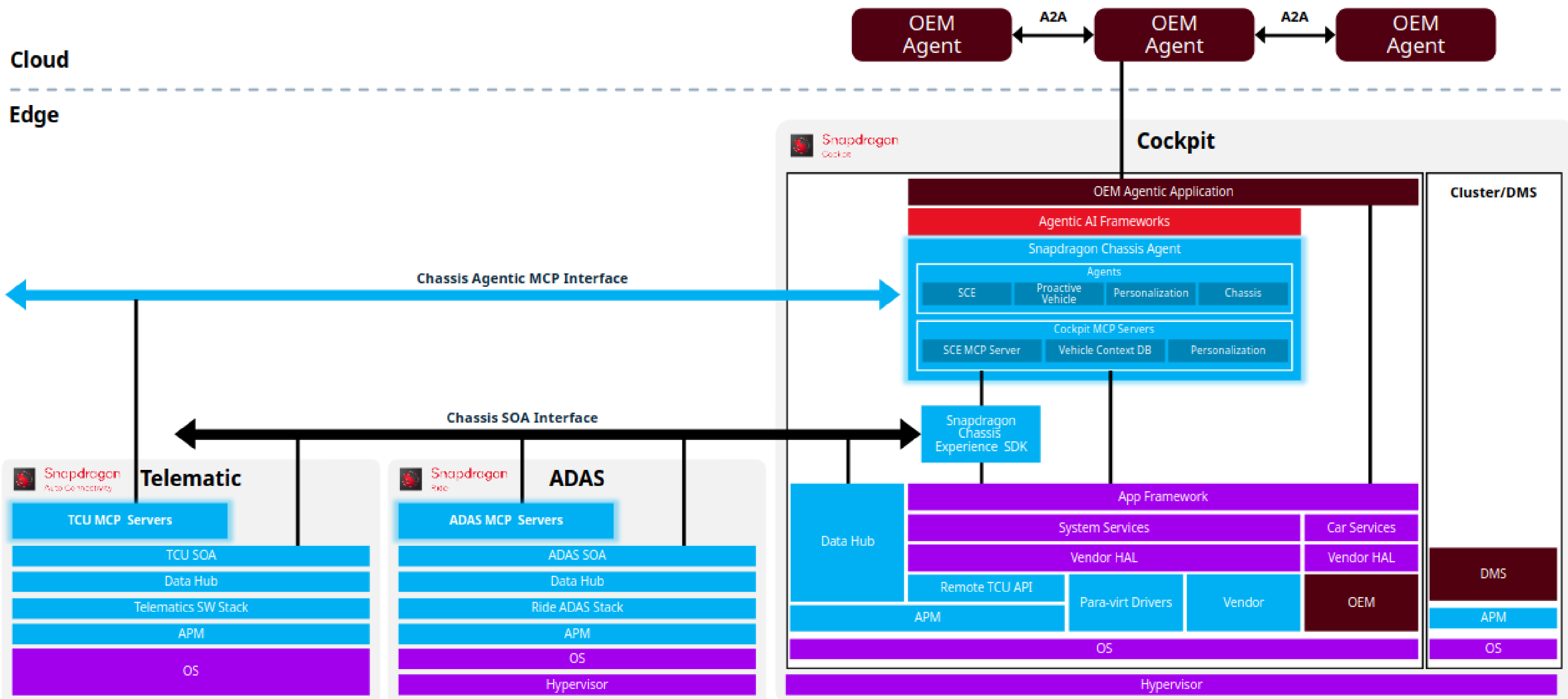

Figure 8: System architecture of Snapdragon Chassis Agents

The architecture is designed to operate both on the cloud and in the vehicle (edge).

Cloud

The OEM agent runs in the cloud, enabling remote diagnostics, updates and advanced analytics. Distributed intelligence and coordination move between edge and cloud and across domains, via A2A.

Edge

At the edge, the architecture is segmented into distinct domains, each with its own stack and middleware.

Telematic

The TCU MCP server in the telematic domain manages connectivity, remote diagnostics and over-the-air updates. This domain uses an SOA, a dedicated data hub for telematics data, a telematics software stack, APM and a Linux operating system.

ADAS

Similarly, ADAS MCP servers control perception, planning and actuation. The ADAS SOA layer enables modular service calls, while the data hub supports real-time analytics and decision making. The Snapdragon Ride ADAS Stack provides advanced driving assistance features, and the domain is underpinned by APM and a QNX hypervisor for safety and isolation.

Cockpit

This domain is the most complex, integrating OEM agentic applications and frameworks that enable agentic behavior and orchestration across all domains.

The Snapdragon Chassis Agent is central to this domain, managing context, personalization, chassis functions and user experience. Components include the Snapdragon Digital Chassis Experience, context and personalization modules, the Snapdragon Digital Chassis Experience SDK and MCP servers, vehicle context database and personalization.

The cockpit domain also includes its own data hub, app framework, system services, car services, remote TCU API, paravirtualized drivers, vehicle hardware abstraction layer

(VHAL), OEM-specific logic and both Linux and Android operating systems. QNX/Gunyah hypervisors enable safety through isolation.

Communication and orchestration

Two interfaces connect the telematic, ADAS and cockpit domains. The Snapdragon Digital Chassis Agentic MCP Interface connects agentic applications and frameworks to MCP servers for control and orchestration of vehicle functions. The Snapdragon Digital Chassis SOA Interface links SOA layers, supporting service calls and data exchange.

Cluster/DMS

Focused on driver monitoring, this domain includes a dedicated DMS system, APM and QNX/Linux operating systems for real-time and safety-critical functions.

Advantages of the Snapdragon Chassis Agent architecture

The architecture takes advantage of multiple data hubs for domain-specific data management, robust middleware for abstraction and orchestration, and hypervisors for mixed-criticality isolation and safety. Personalization and context are managed by the Snapdragon Chassis Agent, enabling tailored experiences for occupants based on preferences and real-time contextual data.

Conclusion

Qualcomm Technologies has been and continues to be in the forefront of providing compute platform and software infrastructure for software-defined vehicles. It is poised to deliver the dominant software and hardware around which OEMs can advance to agentic AI and the AIDV.

Snapdragon Chassis Agents are designed to enhance automotive experiences by integrating complex vehicle systems with user applications. Built upon AI and compute power from Qualcomm Technologies, the agents provide real-time orchestration, context-aware adaptation and proactive personalization across vehicle domains, such as ADAS, cockpit, telematics and V2X. Their hybrid architecture—combining on-device and cloud intelligence—supports robust safety, ongoing feature updates and scalability.

By enabling distributed reasoning and fast decision making, Snapdragon Chassis Agents drive connected mobility forward, offering OEMs modular, security-rich, customizable platforms for smarter, more engaging vehicles.

Appendix: Abbreviations

Acronym or term	Definition
A2A	Agent-to-agent
ADAS	Advanced driver assistance system
AIDV	AI-defined vehicle
APM	Automotive platform middleware
BSP	Board support package
DMS	Driver monitoring system
IPC	Inter-process communication
IVI	In-vehicle infotainment
LLM	Large language model
MCP	Multi-component platform
MLOps	Machine learning operations
MMU	Memory management unit
OEM	Original equipment manufacturer
OTA	Over-the-air
RTOS	Real time operating system
SAFe	Scaled Agile Framework
SDK	Software development kit
SDV	Software-defined vehicle
SLM	Small language model
SOA	Service-oriented architecture
SoC	System on chip
TCU	Telematics control units
TelAF	Telematics Application Framework
V2X	Vehicle-to-everything
VLA	Video language action model
VLM	Video language model