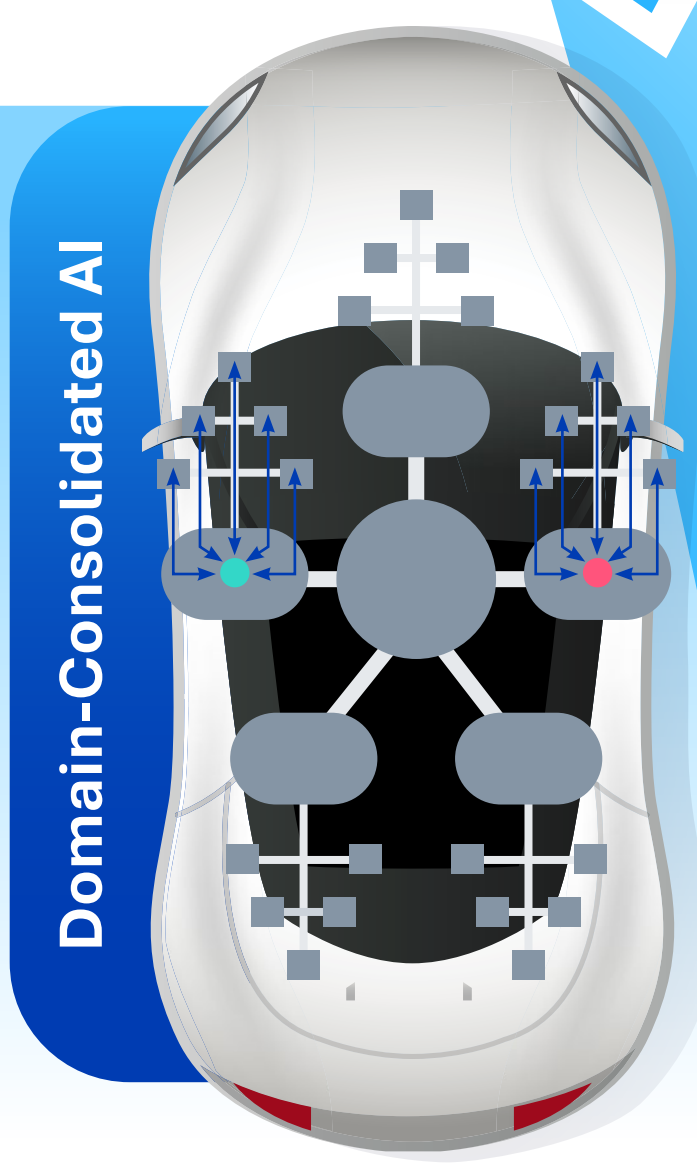
Evolution of AI Integration in Vehicles

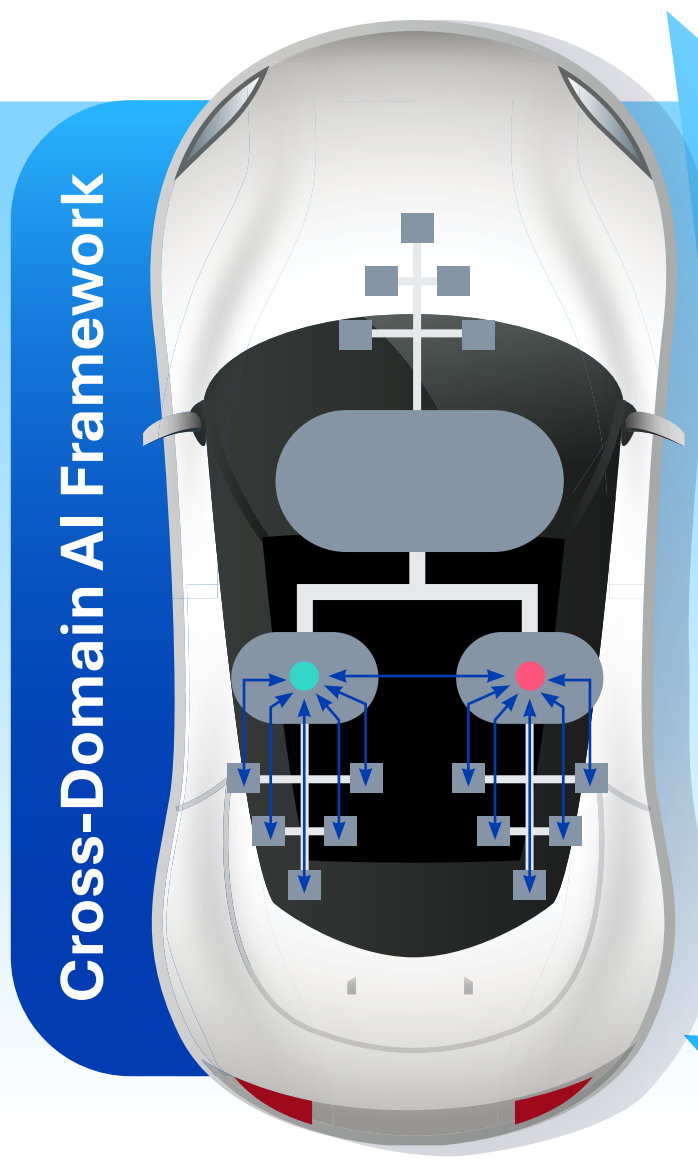
Phase 1: Connect

- E/E Architecture: Distributed
- Computing Resources: Highly constrained (small ECUs)
- Key Characteristic: Isolated AI systems scattered throughout vehicle systems
- Example: Separate modules for parking, lane detection
- Impact: High redundancy, severely limited AI capabilities



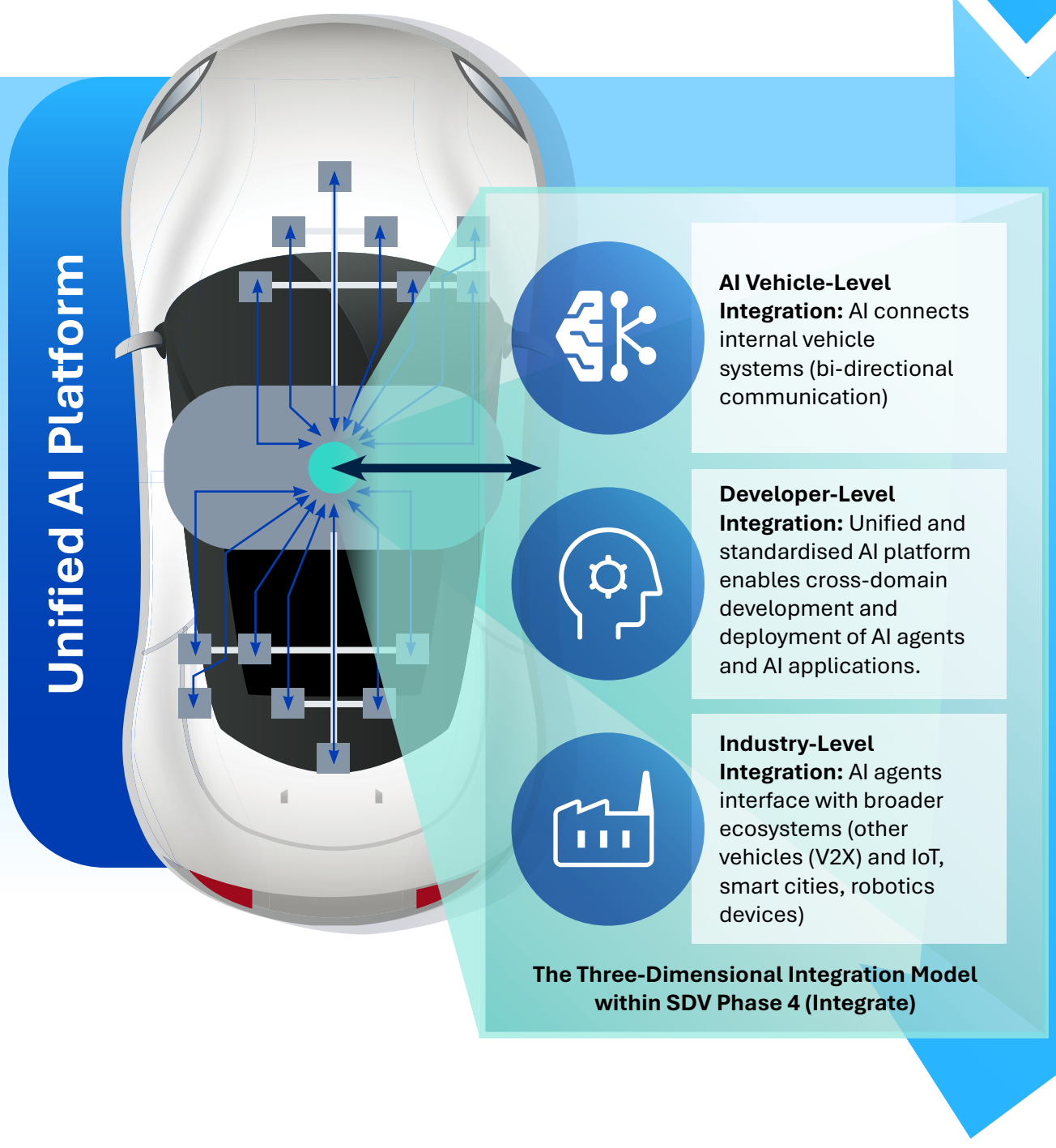
Phase 2: Augment

- E/E Architecture: beginning of domain controller centralization
- Computing Resources: Domain-specific
- Key Characteristic: Unified within domains
- Example: Single AI managing all ADAS functions
- Impact: Reduced redundancy, domain-optimized capabilities



Phase 3: Automate

- E/E Architecture: beginning of zonal implementation/hybrid zonal
- Computing Resources: Shared across connected domains
- Key Characteristic: Multi-domain integration
- Example: Connected cockpit and ADAS systems capabilities, cross-domain diagnostics and performance monitoring
- Impact: Enhanced insights, resource sharing across domains



Phase 4: Integrate

- Architecture: Full zonal architecture/Centralized Computing
- Resources: Dynamically allocated across entire vehicle or even the cloud
- Key Characteristic: AI-Led Software Architecture that has AI as the central orchestrator of vehicle functionality rather than a component
- Example: AI agents with full sensor/actuator access plus cloud and external device integration
- Impact: Maximum efficiency, maximum and flexible insight collection, flexible resource allocation, ecosystem integration, exponential innovation through developer ecosystem

SDV Maturity Phase

Evolution Benefits:

➔ Increasing available computing power ➔ Growing computational flexibility ➔ Expanding AI capabilities ➔ Optimizing resource utilization ➔ Enhancing vehicle intelligence ➔ Enabling ecosystem connectivity

arm

Omdia by informa techtarget