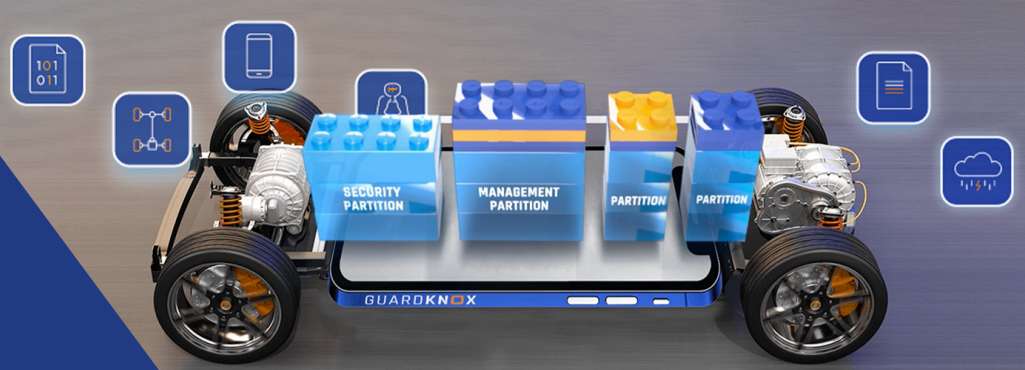


# GUARDKNOX

## SECURE SOA FRAMEWORK



### INTRO

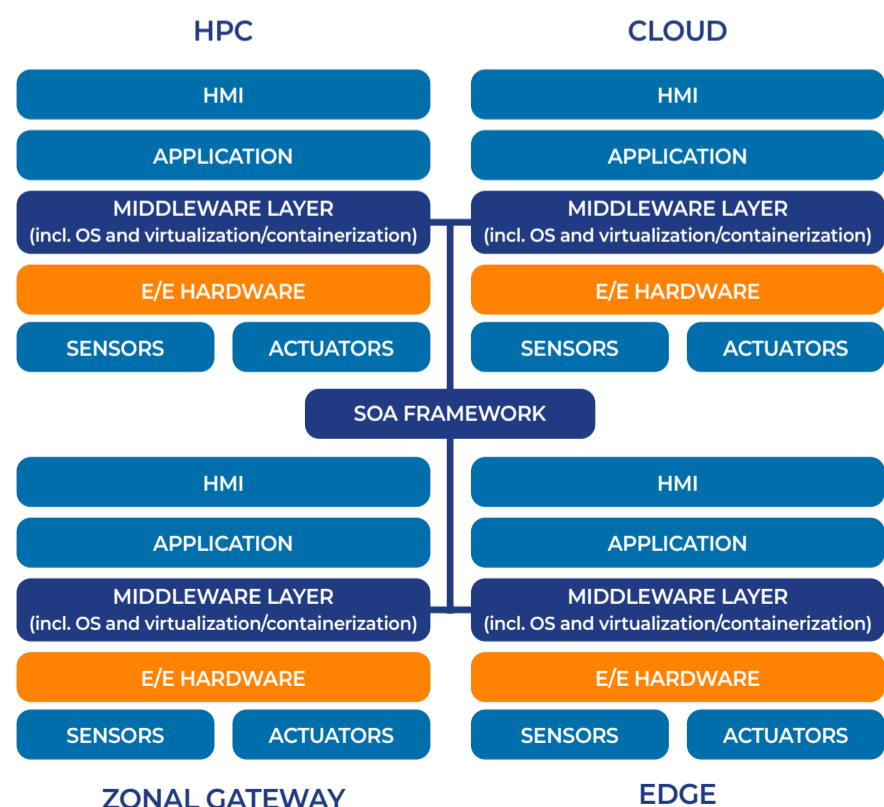
GuardKnox's SOA Framework is a comprehensive software lifecycle management product for the next generation of software-defined vehicles. The framework is domain agnostic and reduces the time to market for new functionality from years to weeks through a dedicated developer tool suite.

### KEY BENEFITS



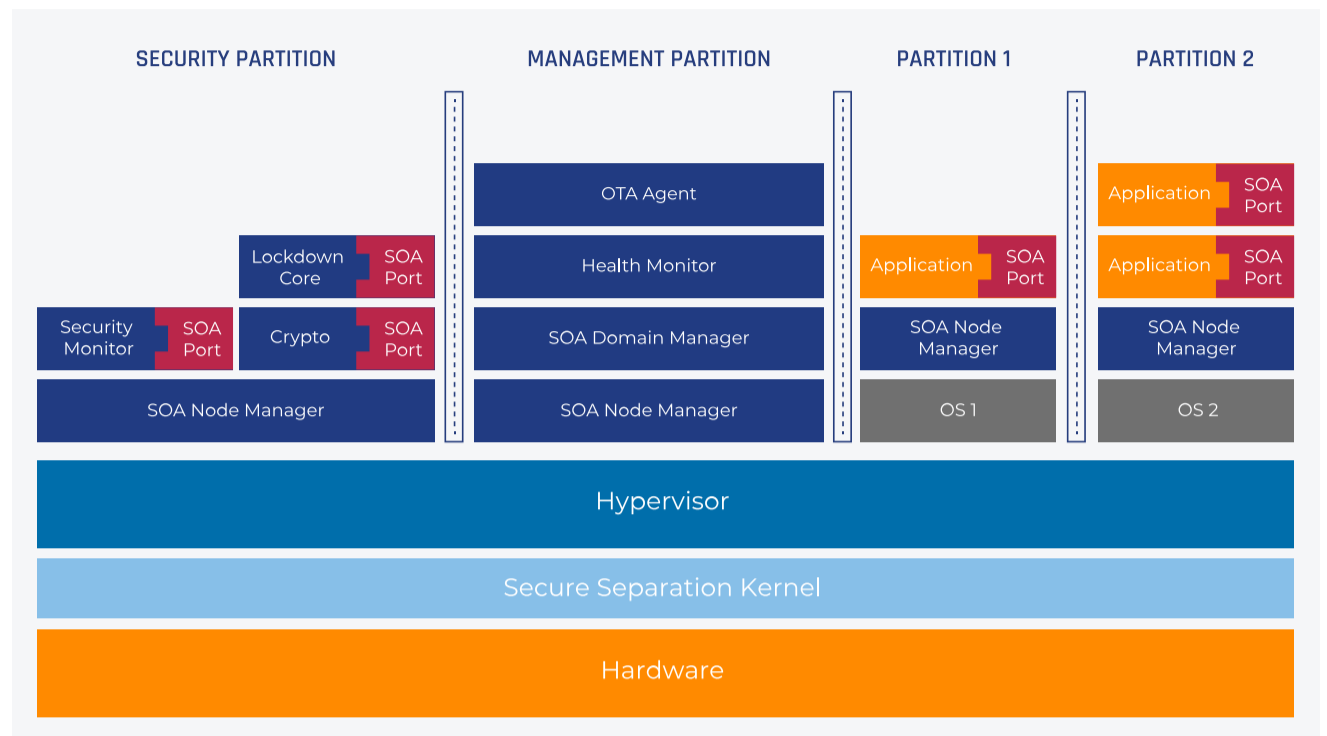
- Provides HW abstraction to decouple the platform and application allowing commonality between different HW platforms and the reuse of the corresponding software components**
  - Enables simplified version management, OTA management and reduction of development costs for the entire vehicle by eliminating dedicated SW solutions for each ECU. This includes the dynamic and optimized deployment mechanism integrated with a secured OTA (by any relevant vendor), as well as the management of SW deployment in heterogeneous embedded (e.g., different HW, RTOS, OS), distributed (e.g., different ECUs/DCUs, etc.) systems (HEDs).
  - Ensures robust middleware with the extendable approach (support for different transport middlewares support of different implementation types.)
  - Enables software driven value-added services from driver assistance and maintenance services to entertainment and convenience applications.
- Creates a secure, manageable, and updatable platform enabling 3rd party application developers via standard APIs** – Enables the establishment of a developer community for all applications to be developed in a virtual, cloud-based environment.
- Enables the seamless and secure integration of cloud-based, edge and hybrid applications and data** facilitating the implementation of value-added services and on-demand features.
- Built on an **established** and **field-proven standard**, therefore development risks are significantly lower.
- Securely manages the entire software lifecycle** while simultaneously monitoring SW health to enable compute load balance through all vehicle computational resources. This reduces the number of HW components while introducing new features during the lifetime of the vehicle.
- Reduces Time to Market for OEMs and Tier-1s significantly through:**
  - SW updates and the implementation of new features and functionalities managed by a model driven development tool suite and code generation/management providing the technical capabilities to implement new revenue streams (e.g., features/services-on-demand).
  - Cross-domain and cross-geographical development and integration across all R&D divisions. Cross-market integration with a variety of software vendors.
  - New ECU software development in 6 months or less.

### ONE APPLICATION ON ALL MACHINES FOR A SEAMLESS VEHICLE INTEGRATION



**The SOA Domain Manager** is responsible for the “SOA Domain” (virtual application space) managing the lifecycle of SWC and their services. The Domain includes multiple runtime environments for applications, each one called a “Node”. The Domain Manager also decides in which Node to deploy every SWC.

**The SOA Node Manager** is respectively responsible for a single “Node” (single runtime environment) and is adapted to execute control operations specifically in this “Node”. It is managed by the Domain Manager which isn’t specific to any runtime environment.



**The SOA Port** is an interface (defined in OMG IDL) realizing a communication broker connecting the SWC to a particular ESB using a user defined transport middleware. It allows for any SWC to create service-based communication ports and define them in a standardized format, regardless of the actual transport implementation. It also provides security, access control, QoS and other communication related capabilities.

## DEDICATED DEVELOPER TOOL SUITE

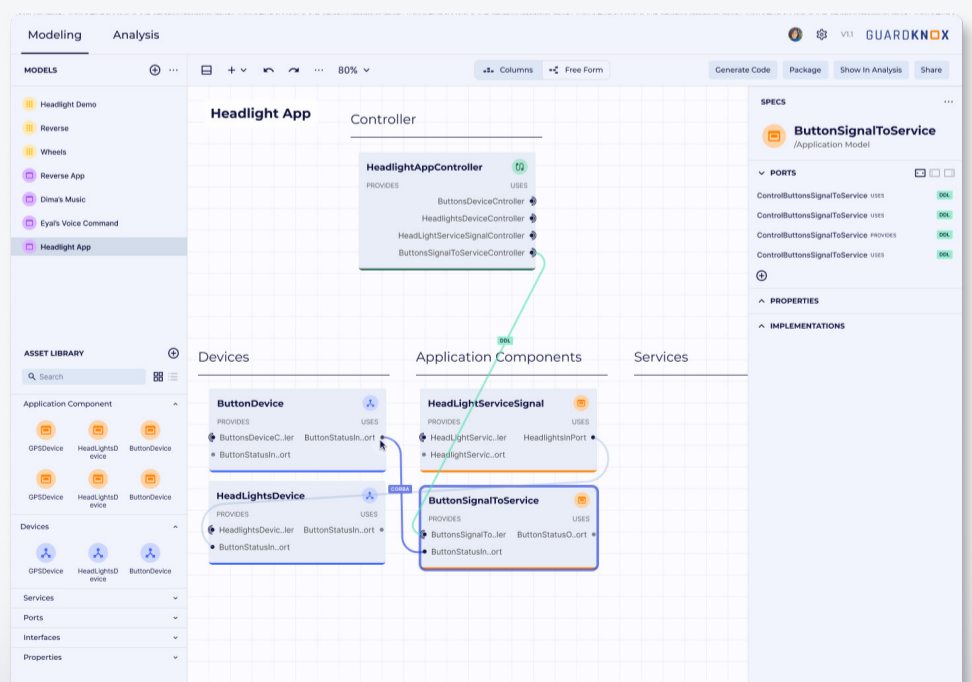
The SOA framework includes **model-based SW development** tools to cover the entire development lifecycle from initial design, development with automatic code generation, through debugging, testing, deployment and post-deployment analysis. The tools are integrated into the Integrated Development Environment (IDE) to streamline the overall development and testing process.

The tool suite consists of:

1. **A modeling tool** for component based development and device abstraction
2. **An analysis tool** for real-time platform integration

These tools:

- Enable automatic code generation for creating well-defined interfaces between devices and applications
- Improve the experience and productivity of architects and SW developers
- Provide intuitive web-based GUIs
- Enable simple deployment for verification and validation during development
- Allow the tester to see the SW status and behavior



Contact us at [info@guardknox.com](mailto:info@guardknox.com) to hear more!