SIBROS

Diagnose Smarter

Making ASAM SOVD Real for the Next-Gen Vehicle

ASAM Regional Meeting North America 2025

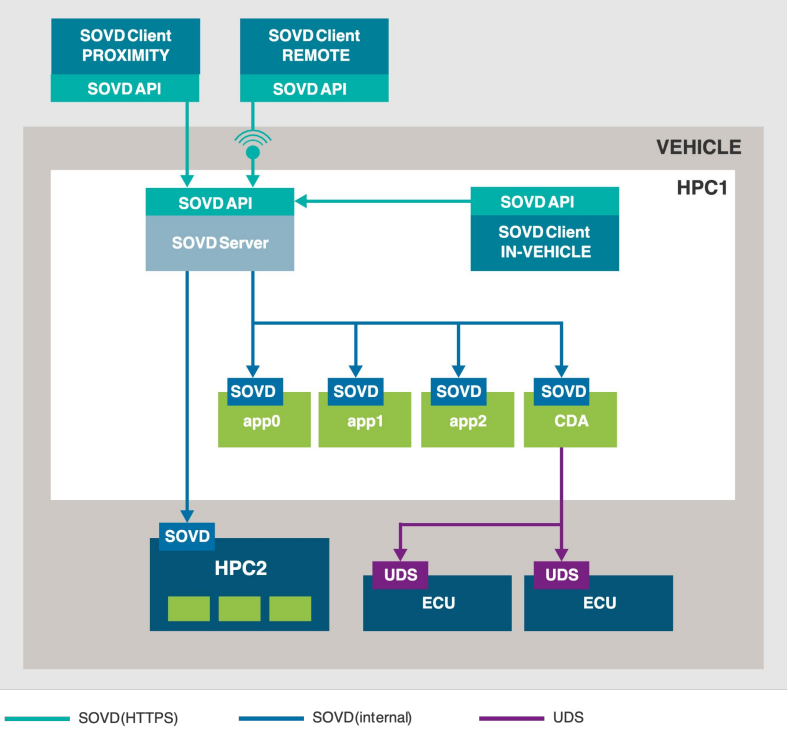
What we'll cover today

1. **SOVD value proposition for OEMs**
   How the introduction of SOVD and evolving vehicle architectures has led to new business opportunities.

2. **From Specification to Implementation**

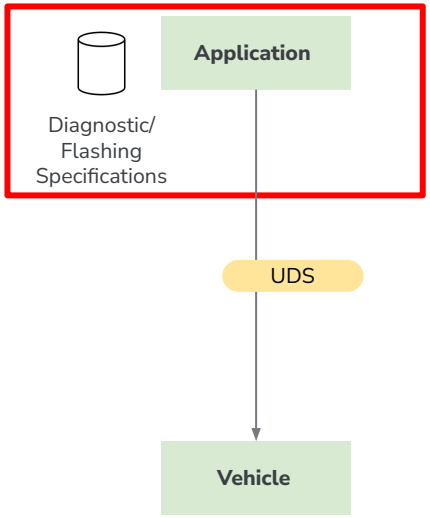
   a. **Design & development considerations**
   Practical challenges around key areas of the standard, especially those flexible to OEMs' implementation.

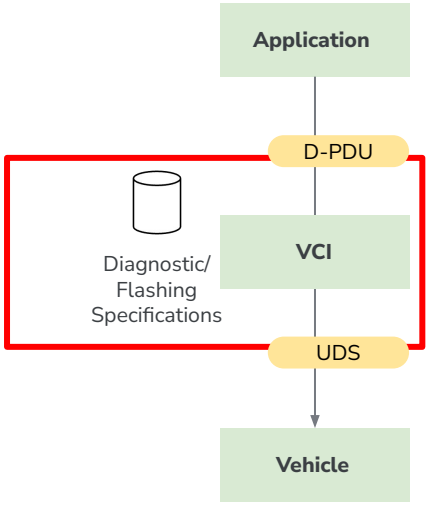
   b. **Production considerations**
   Supply chain, vehicle manufacturing, and change management.



SOVD value proposition for OEMs

Automotive diagnostics and updates yesterday



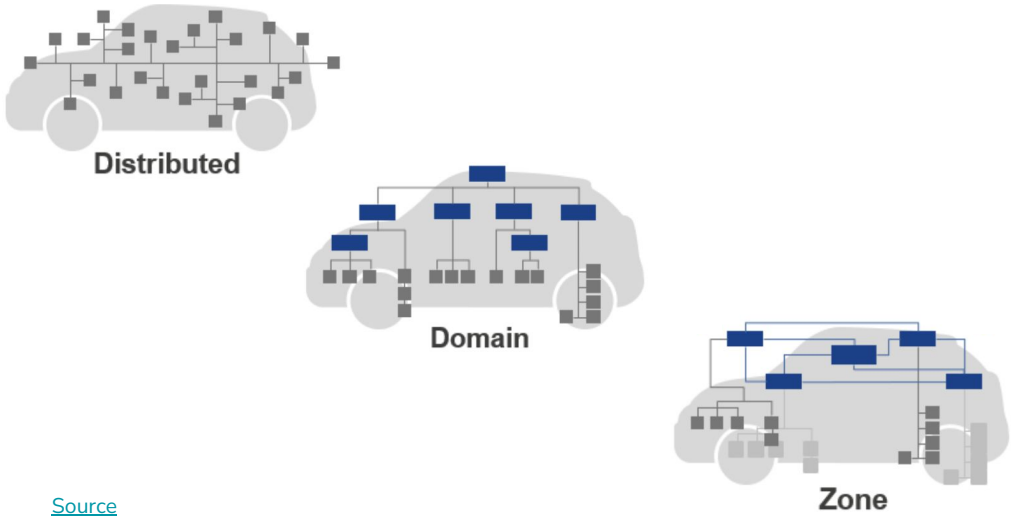


Challenges

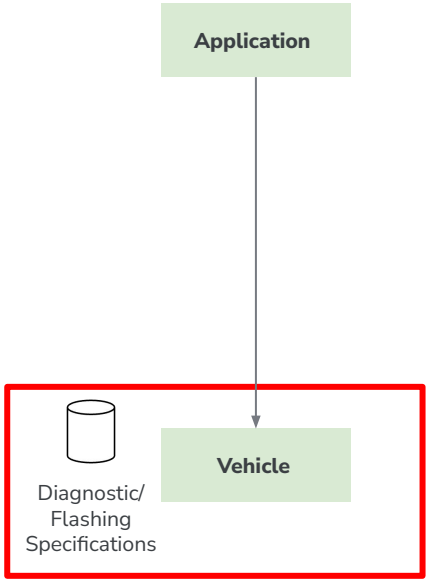
- Offboard specifications
  => **complex applications**
- Change management
- High degree of **customization**

Evolution of the vehicle architecture

- Zonal architectures - **fewer, more capable ECUs**.
- HPCs diagnostics and monitoring **unsupported by UDS**.
- HPCs allow **diagnostic capabilities to be stored onboard** and reported to applications at runtime.

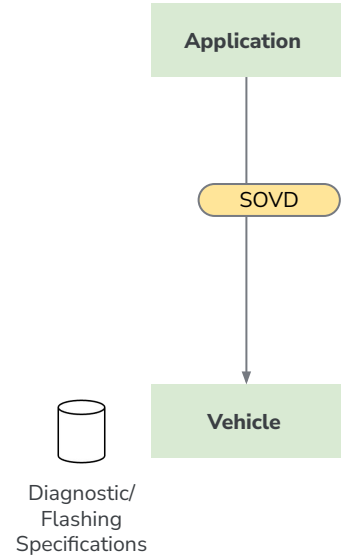


Source



The introduction of SOVD

- A **vehicle diagnostics standard** from ASAM (ISO 17978)
- **APIs** to **diagnose** SDVs, HPCs, apps, and traditional UDS ECUs
- **APIs** to **orchestrate software updates.** Leverages OTA capabilities of a connected vehicle to retrieve and install software.
- Uses **JSON to encode data,** built on HTTP, and follows REST



UDS

Request:
`0x22F123`

Response:
`0x6210312E34352E32313037`

How to convert:
`ALKConfig` —> `0x22F123`
`1.45.2107` —> `0x6210312E34352E32313037`

is specified in the ODX file (diagnostic specification)

What SOVD is *not*

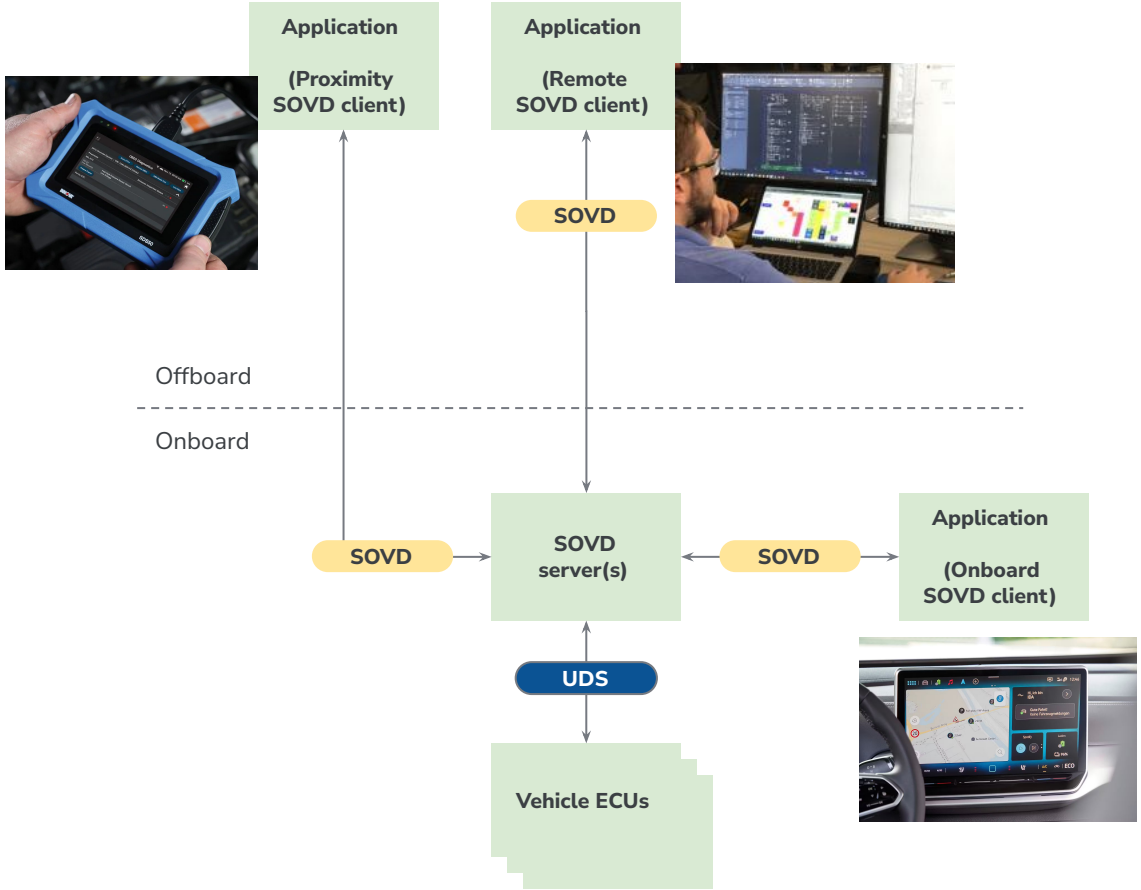
SOVD is not a new programming protocol, unlike UDS

UDS specifies HOW to perform updates
- “negotiate security”
- “transfer bytes”

SOVD goals in summary

Future-proof vehicle diagnostics and software updates

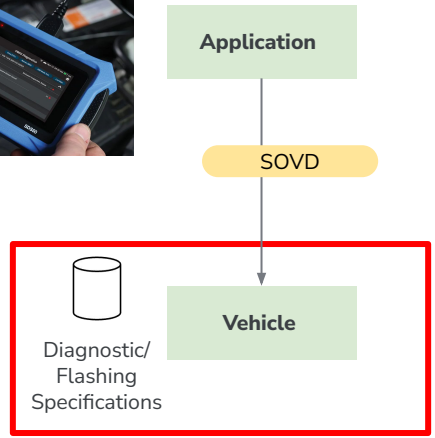
- HPCs and traditional ECUs
- Manufacturing to After-Sales
- Client application simplification
- Standardization across:
  - Remote tools
  - Proximity tools
  - In-vehicle interface



The SOVD Business Case

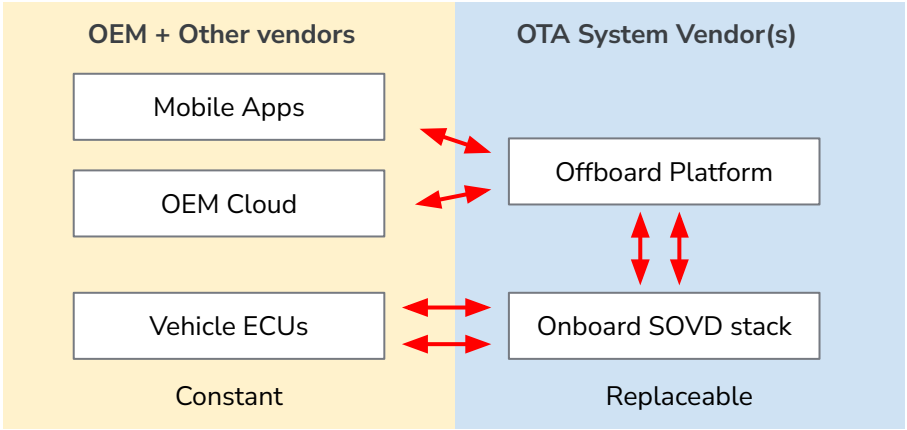
Simplified diagnostic client app development

- Diagnostic and update specifications onboard
- Change management largely abstracted from tooling



Avoid vendor lock-in

- Integration contract persistence despite vendor changes
- De-monopolizes full-stack vendor pricing through separate providers for onboard and offboard software

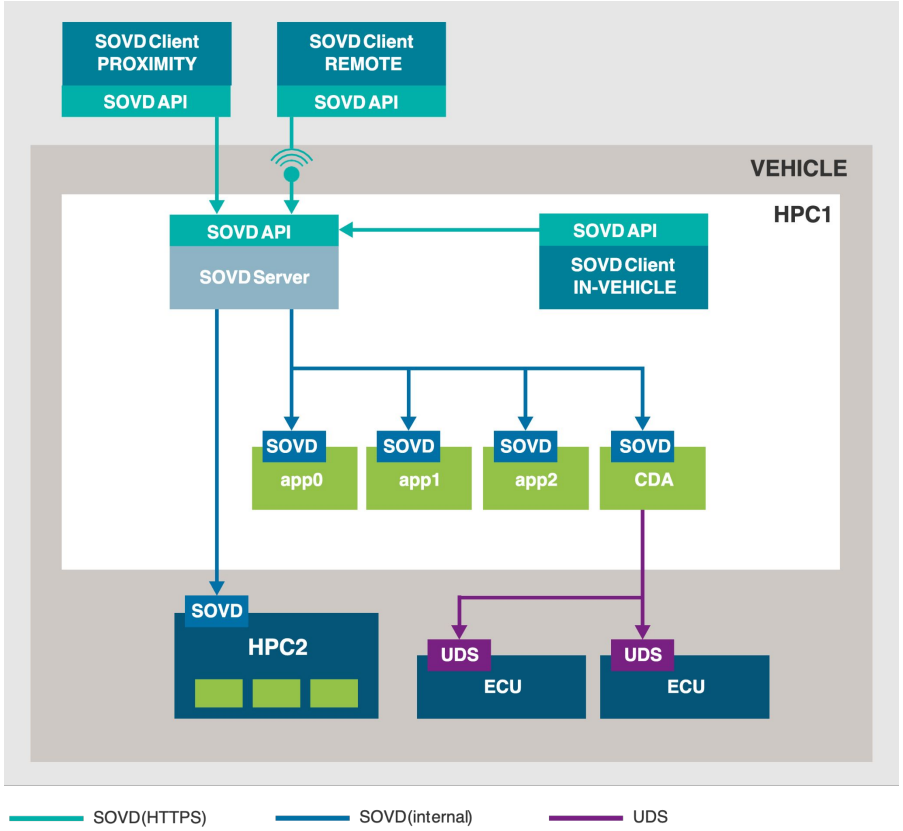


From Specification to Implementation

Design & Development

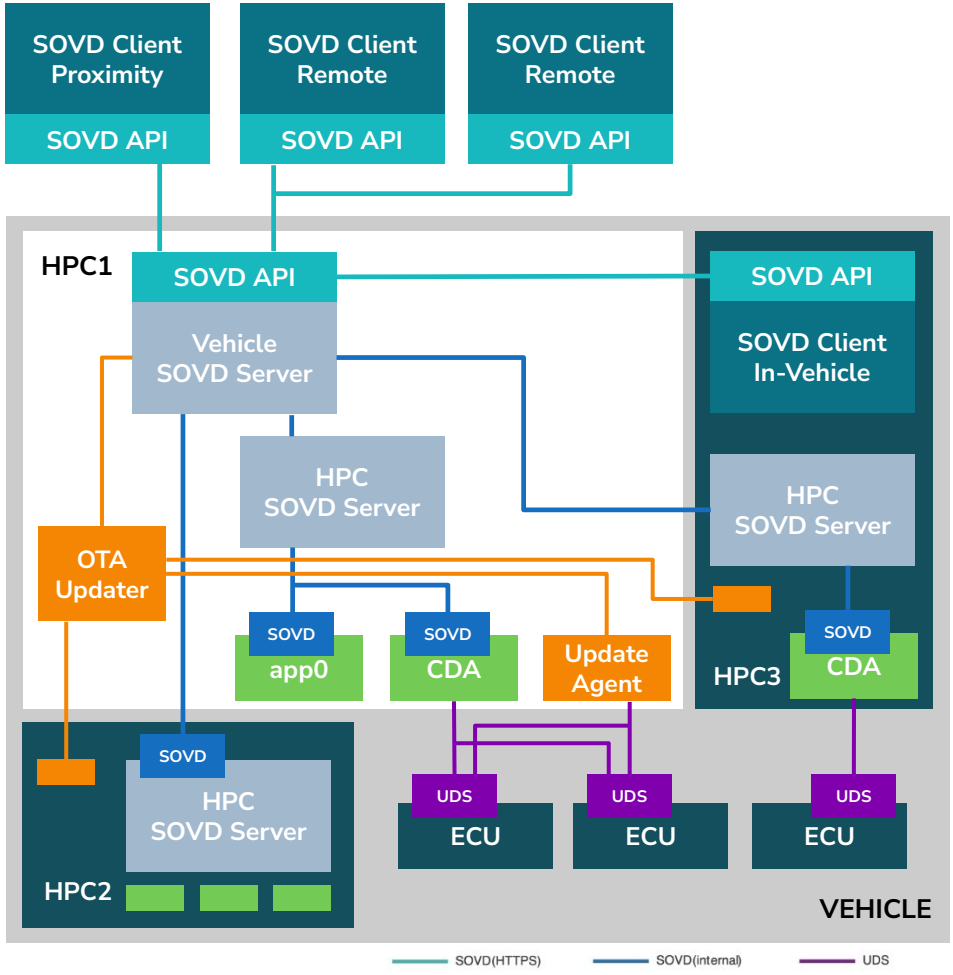
Deep Dive - System Architecture

- SOVD API connects clients to centralized server
  - This interface must be HTTPS
  - Internal interfaces may not be
- Each app on an HPC may be an SOVD endpoint
- Diagnostic Adapter translates SOVD to UDS
- SOVD endpoints span multiple HPCs onboard

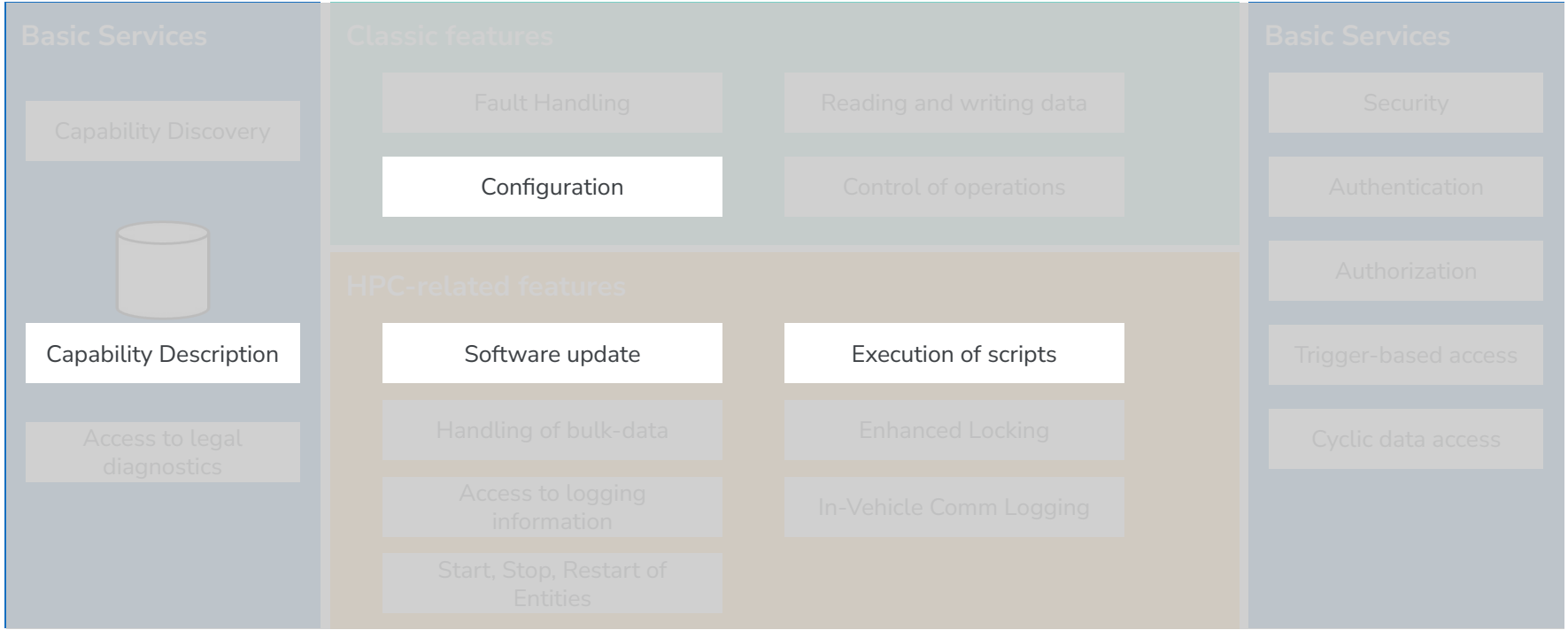


Deep Dive - System Architecture

- More than one remote client connection simultaneously
- Domain-specific HPC server allows for:
  - Subsystem-level use cases
  - Productization of SOVD stack
- Onboard Client on distinct HPC from Vehicle SOVD server
- Reuse of OTA update infrastructure



Summary of ASAM SOVD 1.1 Features



Entities

Entities are associated with diagnostic content. Relationships between them enable representation of entities as the vehicle topology.

Components

- Represent physical (hardware) or virtual (hypervisor) elements
- Run software that may be updated OTA or offline

Apps

- Software that can be added or removed dynamically, given a constant base software

Areas

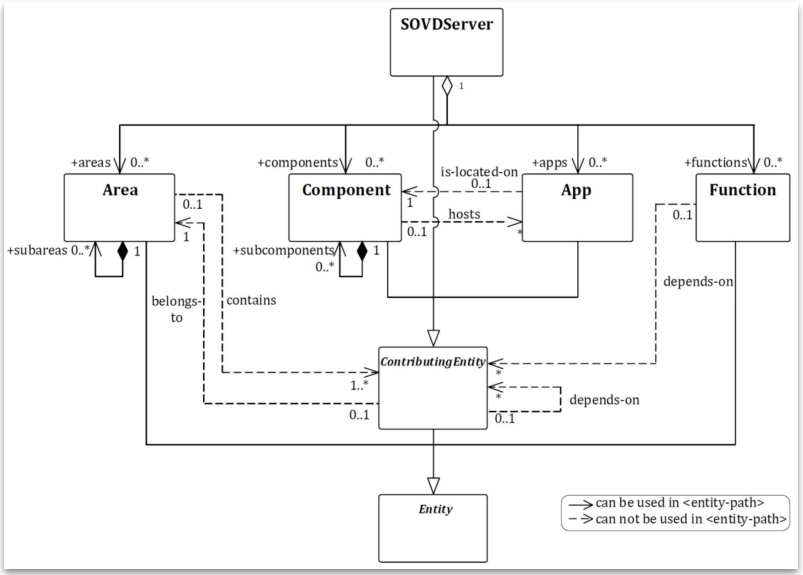
- Logical partitioning by domain or zone to expedite diagnostics

Function

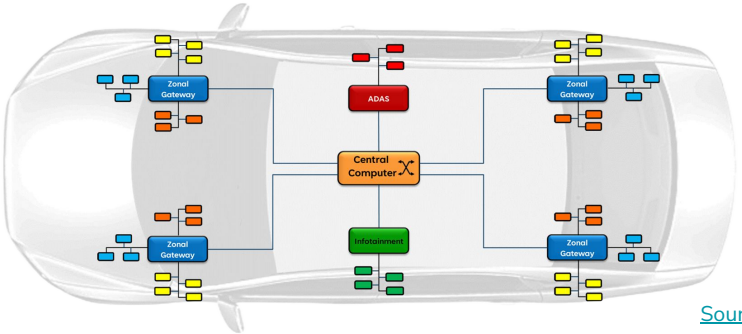
- Vehicle-level operations that are hard-coded in base software on the vehicle

Contributing Entity

- Enables Functions, Components, or Apps to be defined as prerequisites for others. Ex. parts that must be installed together.



Entity Hierarchy



Source

Capability Descriptions

Represents collective diagnostic capabilities: *"How can a diagnostic client interact with the vehicle or its constituent ECUs?"*

- Which methods can be invoked for a given entity?
- Authorization: Clients in which roles can interact with a diagnostic resource?
- Request schema: What should a client input when invoking a method?
- Response schema: How does the client know how to parse the output?

```
"openapi": "3.1.0",
"info": {
  "title": "Vehicle API",
  "version": "1.0.0"
},
"servers": [
  {
    "url": "/"
  }
],
"paths": {
  "/components": {
    "get": {
      "parameters": [
        {
          "required": false,
          "deprecated": false,
          "explode": false,
          "schema": {
            "type": "boolean"
          },
          "name": "include-schema",
          "in": "query",
          "allowEmptyValue": false,
          "allowReserved": false
        }
      ],
      "responses": {
        "200": {
          "description": "",
          "content": {
            "application/json": {
              "schema": {
                "$ref": "#/components/schemas/GetComponentsResponse"
              }
            }
```

Online vs. Offline Description

- **Online**: Dynamic capability discovery for resources as OpenAPI via /docs.
- **Offline**: OpenAPI specification for client reference without dynamic discovery at runtime.

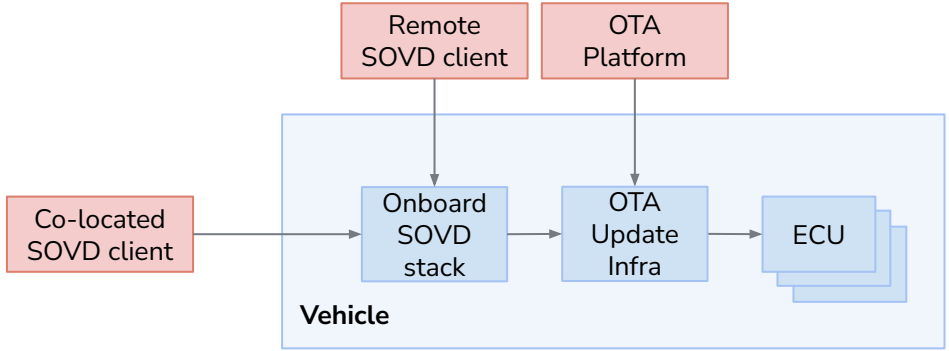
Design Decisions

- Consistency between Capability Descriptions and base software
- Online reporting of UDS capabilities

Updates

SOVD API is used to manage updates triggered OTA or offline.

Orchestration is abstracted to the same onboard infrastructure used for OTA updates, including update sequences & safety checks.



Update Registration

- Definition of target software
- Configuration of update behavior
- Specification of release notes
- Update access (update origin)

Update Preparation

- Download (& staging for A/B ECUs)
- Delta reconstruction

Update Execution

- Installation and activation
- Parameterization i.e. Variant Coding
- Safety checks
- Rollback, if available

Design Decisions

- Simultaneous update registration
- Autonomous packages
- Interaction between OTA and offline updates

Design Decisions

- Automated vs. prepare/execute flow
- USB updates
- Vehicle-specific parameterization

Design Decisions

- Manufacturing vs. after-sales
- Management of update sequences
- Sequencing base software installation and parameterization

Functions

A “functional view” of information; access to diagnostic information spread across several Components.

Server-defined, composite aggregation services.

In the UDS world - Functional Addressing

Example: ***GET /functions/VehicleHealth***

- Custom query parameters
- Custom response body
- Possible reuse of native SOVD methods

Notable aspects

- Stateless and synchronous - No execution lifecycle
- Abstract and declarative - Composite, higher-level
- Server-defined and persistent

Design Decisions

- Consistency between Functions and base software
- Vehicle model- or variant-agnostic nature of Functions

Response Body

Attribute	Type	Description
items	VehicleHealth	Array with the vehicle health of each Component.
schema	Schema	Schema definition of the response.

VehicleHealth Type

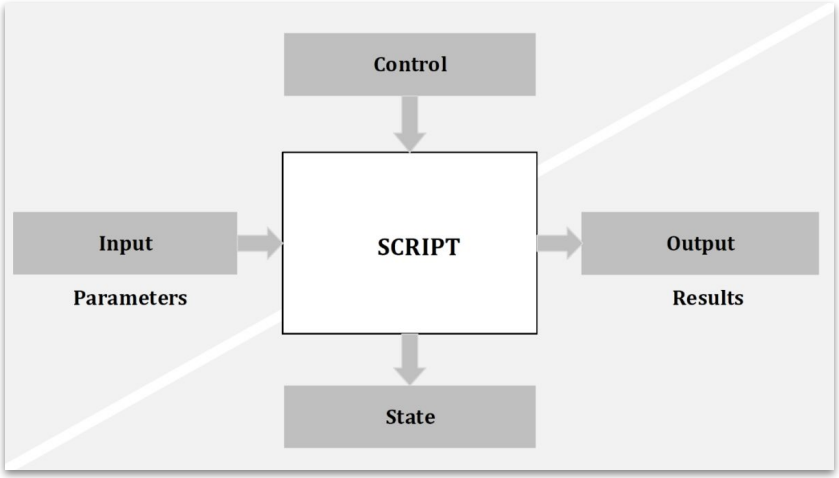
Attribute	Type	Description
component	string:uri-reference	URI to the Component providing the information.
numberOfFaults	integer	Number of faults stored for the Component.
faults	Fault[]	Faults as returned by GET /{entity-path}/faults without setting any of the query parameters.
swVersions	AnyValue	Value of the software version(s). Multiple versions - boot, app, and cal data may be stored.
hwVersions	AnyValue	Value of the hardware version(s).
errors	DataError[]	Error describing why a particular information of the vehicle health of a Component was not retrieved.

Scripts

Scripts represent services and automations that can be executed onboard.

Notable Aspects

- Lifecycle orchestration by the client - upload, start, status, and cleanup
- Not considered vehicle capabilities owing to their dynamic nature
- Script execution runtime and language are OEM-specific
- Serves as a diagnostic extension by reusing SOVD methods and even other OS- or low-level interfaces.



Design Decisions

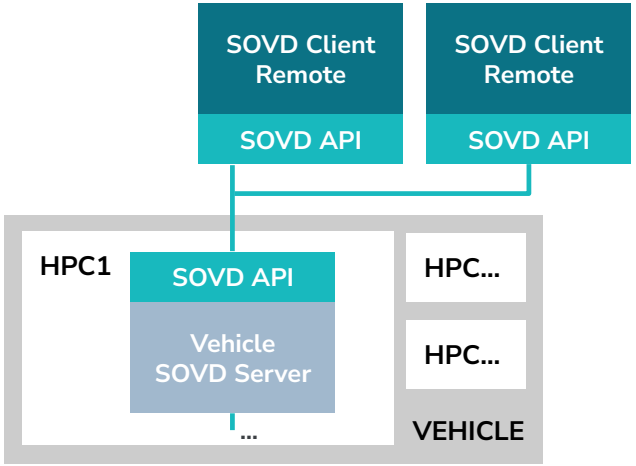
- Scripts vs. Functions
  - Technician involvement requirements
  - Procedural control flows
  - Stateful, asynchronous behavior for long-running flows
  - Workflow stability and ownership
- Runtime choice

Remote Diagnostics

SOVD enables clients to diagnose vehicles via onboard, proximity, and remote channels.

Notable Aspects

- Remote diagnostics APIs are shared with proximity and onboard diagnostics.
- SOVD may be synchronous or asynchronous - requires vehicle to be online in either case.
- Discovery of vehicles via the remote channel is not prescribed by the standard.



Design Decisions

- Remote diagnostics via cellular and WiFi
- Onboard positioning of SOVD server vs. connectivity gateway
- Secondary authentication and authorization offboard
- Synchronicity and payload sizes over the air
- Peak volume of simultaneous vehicle diagnostics connections
- Shared development of proximity and remote solutions

Production

Vehicle Assembly and Supply Chain

Seamless supply chains

- Vehicle Model- or Variant-agnostic ECU configuration to eliminate inventory management overhead
- Subsystem-level quality testing from Platforms sourced from suppliers
- Temporary certificates to test client/server connection security

Flexible, fast, and secure vehicle assembly

- Vehicle onboarding at OEM plant
- Automate to the extent possible
- Proximity diagnostics & updates to avoid wireless latency bottlenecks
- Option-based variant- or customer-specific vehicle configuration
- OEM certificates to guarantee authorized access to the vehicle post-sales

After-Sales Service

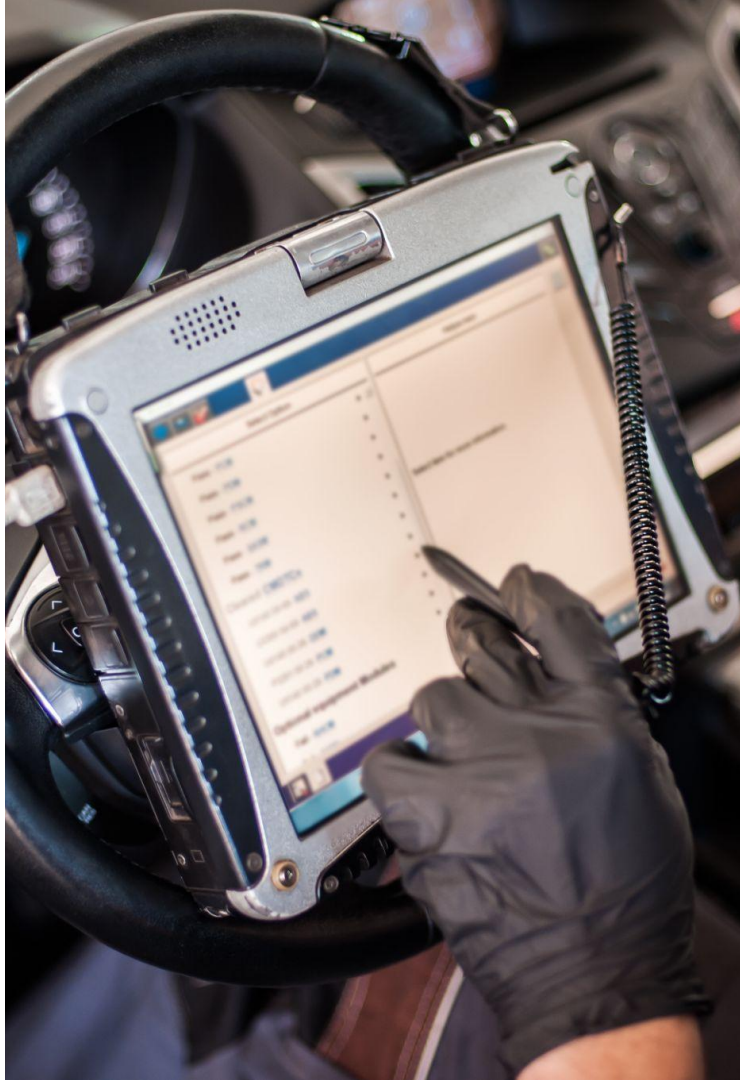
Remote Assistance

- Remote client registration with OEM ecosystem
- Debug log retrieval and high-resolution data logging configuration

In-Shop Service

- Restricted access to updates that require technician intervention
- Part replacement and resynchronization of software configuration
- Re-registration with the OEM Platform

Onboard Diagnostics



Let’s connect!

SOVD consulting, product suite, or services

https://www.sibros.tech/contact