

VSS Datapoints

Mobility Cloud

Node.RED

Android

"APPS"

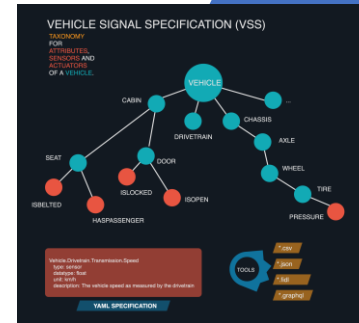
DAPR.io

IoTEA

VISS



KUKSA .val

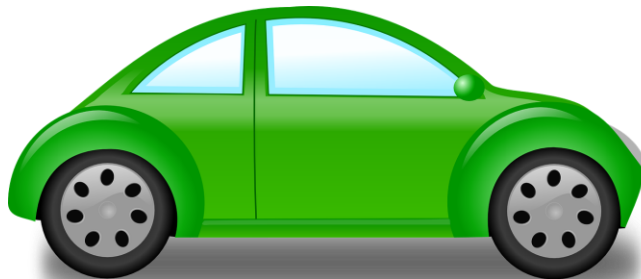


Wisdom

Knowledge

Information

Data



# Why VISS (V1) was a good choice

- On VehicleComputers (processor) we do not want a “linkable” API, but a networked one
  - Suitable across sandboxed domains: E.g. containers/VMs
  - Suitable across ECUs
  - Can be “easily” protected
- It is sort of a standard
  - Always better to say “Hey, it is W3C VISS, check it here”, than “Look at the code to see how it roughly works”
- ASCII/JSON based
  - Easy to use from high-level scripting
- Pub/sub support

# Gaps from current Kuksa.val PoV

- Websocket is nice since it can support Pub/Sub as well as query/reply, but tooling is not as easy/ubiquitous than e.g. REST
- When another “framework” (e.g. IoTEA) is on top, maybe want a “harder/smaller” interface for performance Reasons (e.g. GRPC) (i.e. plumbing API instead of porcelain API)
- VISS V1 in parts a little under-specified