

20200514-Virtual-Technical-Summit-CCS-Workshop-Minutes

1	## Introduction
2	<p>Philippe Robin welcomes everyone and hands over the mic to Kevin Valdek</p> <p>50 attendees</p> <p>slide deck for this workshop is here</p>
3	## Project overview
4	<p>Kevin: this is the presentation Gunnar and I delivered in the Automotive World webinar last week, recording is available</p> <ul style="list-style-type: none">• Introduction<ul style="list-style-type: none">◦ slide #4: Project charter<ul style="list-style-type: none">▪ Even though cars are connected, they are limited to specific OEM/brands▪ A lot of ideas and start ups introducing new services...▪ We want to harmonize these activities... Define building blocks with flexibility for common solutions... Define data exchange and access◦ slide #5: Vehicle data<ul style="list-style-type: none">▪ not limited to specific types but clustered by anonymity (personalised, pseudonymized, anonymized)◦ slide #6: Services / Needs<ul style="list-style-type: none">▪ up-to-date data, typically telematic through a cloud server, but also bulk on big data• Data Model<ul style="list-style-type: none">◦ slide #9: Common data model<ul style="list-style-type: none">▪ the industry needs this (different implemenations for each OEM... This is to avoid this situation)▪ Gap analysis in the CCS project (CVIM, SensorIS, ISO20078, Android Automotive, VSS) in 2019◦ slide #10: Vehicle Signal Specification (VSS)<ul style="list-style-type: none">▪ VSS is at the core of the CCS project◦ slide #16: Value exchange formats<ul style="list-style-type: none">▪ proposed protocol to group and exchange VSS data• Framework:<ul style="list-style-type: none">◦ slide #18: capture data in vehicles<ul style="list-style-type: none">▪ ECUs speaking VSS2 (SOME/IP, DDS, HTTP/REST) or current bus translation◦ slide #19: cloud transfer & storage<ul style="list-style-type: none">▪ Transfer/Storage with the W3C defined protocol Gen2, "Gen2" is a vehicle API over HTTPS and WebSockets + data lake based possibly on object storage in a later stage▪ Buffering (high frequency, poor connectivity...) to take into account◦ slide #22: Neutral servers and data marketplaces opportunity included (in accordance to ISO20078) or simple 3rd party access
5	<p>discussion on framework architecture</p> <ul style="list-style-type: none">• Iyyaz Baber: What about a noSQL DB in the cloud?• Kevin: We could. In the end, the proof-of-concept came from our experience with relational DB, there is no restriction• Ulf: the server exposes an API. As long as it follows the API, the DB could be of different types.• Gunnar: this would depend on the data (time series fitting into a relational DB ?).• Teddy Zhai: When you say: "Cloud is Client, and in-vehicle is Data Server", do you mean in a logical view or technological view? It means, for example, Data Server needs to open ports and listen on them.• Ulf: it's both, the data servers needs to open ports and listen, and response to requests (standard client-server model)• Gunnar: it can vary from this current proof-of-concept architecture to a production-ready solution. Typically there could be an SSL tunnel set up by OEMs• Teddy: This is a major architecture decision in our view. It would be great if GENIVI can provide a threat model for it.• Gunnar: I think in the later stages of proof-of-concept development, it is worthwhile to address these questions about how to set up private networks between some parts (e.g. OEM to car). If we end up using SSL/TLS then the GENIVI Security team has previously published a best practice document for SSL/TLS which avoids a lot of security pitfalls.• François Fischer: data model - you can add SAREF/Autopilot• Philippe: we had a look at SAREF proposal in September 2019 (SAREF Automotive - https://www.w3.org/2019/09/trans-data-ws/SAREF.pdf) and estimated that it was more on the long term (5+ years) goals, it might be worth rechecking the status of SAREF work• Francois: I coordinated a project about IoT and automated driving, i.e. AUTOPILOT and we pushed data models to SENSORIS and SAREF, see https://autopilot-project.eu/• Gunnar: there are datasets included in the current VSS work, we expect to use an Electric Vehicle (EV) use case and data set for the proof-of-concept demo implementation• Gunnar: our objectives are to get fast results, identify technologies and components that need to be standardized
6	## proof-of-concept demos

7	<p>Demo #1 - OEM cloud</p> <ul style="list-style-type: none"> • Ulf: shows a Cloud and Connected services demo involving 3 components: <ul style="list-style-type: none"> ◦ data server (W3C Gen2), ◦ data base (Open Vehicle Dataset Server a.k.a. OVDS) ◦ CCS vehicle client • Data server can be accessed through multiple protocols (HTTP, WebSockets) with VSS(2) data (but open to use other compatible domains - with a service manager) • Data base is a SQL database with a VSS-to-DB adapter • CCS client sits in between and creates a list of all VSS leaf nodes and issues read/write requests to the Gen2/OVDS servers
8	<p>Demo #2 - Vehicle data generation</p> <ul style="list-style-type: none"> • slide deck is here • Stefan Vysocki (from the GENIVI AASIG Vehicle Data / VHAL project) shows a demo of a vehicle data feeder using a vehicle simulator and a GraphQL server <ul style="list-style-type: none"> ◦ simulator used in the demo is OpenDs, a frame example is given (could be VSS too) ◦ the demonstration includes the simulator interface and a GraphQL server to retrieve values from the simulator • Kevin Valdek: Could we use a fleet instead ? • Stefan: Yes, by configuring multiple ports • Stephen Lawrence: Could you containerize the simulator and automate tests ? • Stefan: It is partially containerized, no issues.
9	<p>Demo #3 - GraphQL client on top of a VSS schema</p> <p>slide deck is here, Kevin shows a demo</p> <p>Discussion</p> <ul style="list-style-type: none"> • How to have data anonymization ? • there is consent here • What about the identification? • Here we could have many types of identifiers, such as what a brand allows you to access (a certain fleet for instance) • What happens when the user removes consent (need to do something on the 3rd party side) ? • These are great use cases, we should look into them. • How do we categorize what is to be anonymized and what is not ? • Since privacy is partially region-based, it might be better to have configuration outside of the VSS
10	<p>Discussion on technology options</p> <ul style="list-style-type: none"> • Gunnar introduces the topic with this slide deck • How to generate data <ul style="list-style-type: none"> ◦ Usefulness of an easy to run driving simulator <ul style="list-style-type: none"> ▪ it would be nice to use the simulator as a driving game and generate data ▪ Iyyaz: I go for it ! ▪ Gunnar: which operating system would you use ? ▪ Iyyaz: ubuntu, linux or windows 10 ▪ Christian: having vehicle data would be useful to test our APIs ! ▪ Stephen Lawrence points out that CAN can-utils has the basic cansend to send can msgs, look at https://github.com/linux-can/can-utils, and https://manpages.debian.org/stretch-backports/can-utils/cansend.1.en.html ▪ How important is the sanity of the data ? Could we do statistical analysis on random data? What about randomly generating data ? ▪ Gunnar: Is that sending a random generated data you mean? ▪ Stephen: can-utils also has cangen for random data ▪ Stephen: it seems also that candevstudio may have some generation capability, look at CANdevStudio repository • Alternative technologies <ul style="list-style-type: none"> ◦ Why not using MQTT between the vehicle and the cloud (rather than other than W3C Gen 2) ? with MQTT there is a need to define the payloads ◦ using maybe a NoSQL data base ◦ Abhijeet: mentions the use of a replica of the car like AWS Car Shadow and like the Ditto digital twin - Device as a service: Ditto project from Eclipse https://www.eclipse.org/ditto/ • How to do high-volume tests ? • Evaluating the performances ? • What about security ?