#### VSS – A deeper Dive –

How to build things with VSS - An example

Sebastian Schildt, Erik Jägervall, Adnan Bekan COVESA AMM, September 25<sup>th</sup> 2024

Updated Q3 2024 edition



1.1.1

#### NOTE

#### **NOTE**

There are some command lines in this presentation. It seems when copying them, sometimes some whitespaces are converted not correctly so you get weird error messages like missing arguments. Somewhere "near" this presentation you should have found an archive that contains safely copyable commands.



# In the past...

... or earlier at this AMM



Where to use VSS?

What of VSS do I need to support?

What is VSS even?

Which software can I use?

I am confused





Wherever you want to!

Whatever you want to!

Amazing. It is not an API, Serialization or protocol though.

Many!

JOIN COVESA



# This changes now

We show you an **example**<sup>1,2</sup> of how to use VSS

#### <sup>1</sup> We will...

- Iook at and modify VSS files
- apply the VSS overlay concept
- use VSS tools
- use real CAN Data
- write real code interacting with VSS

<sup>2</sup> We will not...

- look at every tool or framework supporting VSS
- show all domains (not too much cloud in here)



# The idea







# VSS – We use VSS already, are we covered?

🔹 💿 🗈 👻 < 🚿 🦉 🚍 😰 🔒 github.com/COVESA/vehicle_signal_specification/rel	elee 940 Č + Č	5
COVESA / vehicle_signal_specification	Q   + • O n @ 🍪	
↔ Code ⊙ Issues 53 In Pull requests 7 □ □ Discussions ⊙ Actions	ns 🖽 Projects 🖽 Wiki 🛛 …	
Releases / v4.1		
v4.1 (Latest	Compare 🗸 🖉 🗍	
erikbosch released this Dec 22, 2023 39 commits to master since this release	© v4.1 -≎ c7eec98	
Changes with this release		
Unit file syntax updated, Quantity file introduced		
A new syntax for unit files is introduced. Old syntax still supported. It is also possible to define quantity files, and a default quantity file ( quantitie has been added	es.yaml)	
New signals		

layground.digital.au' a Search			vO.1 IN Prototypes	=
OVESA VSS 🐱			List View	Tree Vi
Q Search	= Filter			
Recen	tly viewed APIs			
Vehicle	BRANCH	. •	0	
Vehicle.ADAS	BRANCH		SPA .	
Vehicle.ADAS.ABS	BRANCH			
Vehicle.ADAS.ABS.IsEnabled	ACTUATOR			
Vehicle.ADAS.ABS.IsEngaged	SENSOR		C D:	Constant
Vehicle.ADAS.ABS.IsError	SENSOR	Vehicle.Speed	Sea Discussion (0)	SENSO
Vehicle.ADAS.ActiveAutonomyLevel	SENSOR	C Tage		
Vehicle.ADAS.CruiseControl	BRANCH	V laga		
Vehicle.ADAS.CruiseControl.IsActive	ACTUATOR	VSS Specification		
Vehicle.ADAS.CruiseControl.IsEnabled	ACTUATOR	DataType	float	
Vehicle.ADAS.CruiseControl.IsError	SENSOR	(Output)		
Vehicle.ADAS.CruiseControl.SpeedSet	ACTUATOR	Description	Vehicle speed.	
Vehicle.ADAS.EBA	BRANCH	Туре	sensor	
Vehicle.ADAS.EBA.IsEnabled	ACTUATOR	Unit	km/h	
Vehicle ADAS EBA Is Engaged	SENSOR	111115		C Ren

# Vehicle.Speed Vehicle.Chassis.Axle.Rowx.Wheel.L/R.Brake. ??? No temperatures

<u>https://github.com/COVESA/vehicle\_signal\_specification/releases</u> <u>https://digitalauto.netlify.app</u>



6

25 September 2024 Copyright © 2023 COVESA

### **Convert VSS model**

	1		
	✓ vehicle_signal_specification		Good to know
EXPLORER Using VSS- tools rewrite tools rewrite > Body > Body > Cabin > Chassis > Driver > Identifier > Identifier > include > OBD > Powertrain > Vehicle ! units.yaml	<pre>! VehicleSignalSpecification.vspec × spec &gt; ! VehicleSignalSpecification.vspec &gt;     # This top to ever vehicle specification that of     # files into a complete specification that of     # files into a complete specification that of     # #     # #     # #     # #     # The vehicle branch for highlevel vehicle s     #     Vehicle:     // Vehicle:     // type: branch     description: High-level vehicle data.     # Include the Vehicle/Vehicle.vspec file and     # #     # #     # #     # #     # Include the Vehicle.vspec Vehicle     # #     ##     # #     # #     # #     ##     ##     # #     # #     # #     # #     # #     ##     ##     # #     ##</pre>	<pre></pre>	<ul> <li>COVESA vss-tools convert .vspec files to a variety of other formats (json, cs, ddsidl, etc.) for use in different tech stacks</li> <li>You can easily write your own converters</li> </ul>
VehicleSignalSpecification.vspec     > vss-tools     S	33 # 34 V	Helpful	commands
<ul> <li>.gitignore</li> <li>.gitmodules</li> <li>.pre-commit-config.yaml</li> <li>CHANGELOG.md</li> <li>CONTRIBUTING.md</li> <li>LICENSE</li> <li>Makefile</li> <li>OUTLINE</li> <li>TIMELINE</li> <li>Prelease/4.0          <ul> <li>\$\$ release/4.0              </li> <li>\$\$ 0</li></ul></li></ul>	<pre>PROBLEM INF0 INF0 ation/st INF0 ion.vspt INF0 INF0 INF0 (amm) st</pre>	ss-tools -vspec ification/spec/Vehicles	SignalSpecification.vspecoutput



# Add additional signal in overlay

! VehicleSignalSpecification.vspec	! dbc_overlay_simple.vspec ×	Good to know
<pre>amm &gt; ! dbc_overlay_simple.vspec &gt; 39  # SG_ BrakeTempRR3FE : 30  40 Vehicle.Chassis.Axle.Row1. 41  datatype: float 42  type: sensor 43  description: Brake Tempe 44 45  Vehicle.Chassis.Axle.Row1. 46  datatype: float</pre>	<pre>{} Vehicle.Chassis.Axle.Row2.Wheel.Right.Brake.Te 10@1+ (1,-40) [0 0] "C" Receiver Wheel.Left.Brake.Temperature: rature FL Wheel.Right.Brake.Temperature:</pre>	<ul> <li>You can <i>add</i> signals to any VSS model using an overlay</li> <li>An overlay resides in seperate files, so this a plus managing (composable) VSS models in your development workflow</li> </ul>
<ul> <li>47 type: sensor</li> <li>48 description: Brake Tempe</li> <li>49</li> <li>50 Valiale Chassis Aula Dav2</li> </ul>	rature FR	
50Vehicle.Chassis.Axle.Row2.51datatype: float52type: sensor53description: Brake T5455Vehicle.Chassis.Axle.R56datatype: float57type: sensor58↓description: Brake T59	Wheel.Left.Brake.Temperature: Hel bec2json.py -o overlay_simple.vspec	pful commands ./vss_rel_4.1.yaml myextvss.json



# Use VSS in-vehicle with KUKSA

	amm — sebastian@docker: ~ — pipenv shell > docker — 104×32		
			Good to know
	ii ii .a" .p"") ii ii .a" .p"") .ent CLI	•	KUKSA is an in-vehicle server for VSS signals It is an Eclipse SDV Open Source project under Apache- 2.0 license (commercial support available)
Default tokens directory: /kuks	a-client/_internal/kuksa_client/kuksa_server_certificates/jwt		
Connecting to VSS server at 127 TLS will not be used. INFO 2024-04-05 09:33:53,241 ku re connection! INFO 2024-04-05 09:33:53,241 ku gRPC channel connected. [Test Client> setValue Vehicle.S	7.0.0.1 port 55555 using KUKSA GRPC protocol. Mksa_client.grpc No Root CA present, it will not be possible to use a s Mksa_client.grpc.aio Establishing insecure channel Speed 42	ecu	
Test Client> getValue Vehicle.S	Helpful c	om	mands
<pre>{     "path": "Vehicle.Speed",     "value": {         "value": 42.0,         "timestamp": "2024-04-0     } }</pre>	docker run -itrmnet=host ghcr.io/eclipse-kuksa/kuksa-databroker:latest docker run -itrmnet=host ghcr.io/eclipse-kuksa/kuksa-python-sdk/kuksa- client:latest		
Test Client>	docker run -itrmnet=host -v \$(pwd):/d databroker:latestvss /data/myvssmodel.js	ata on	a ghcr.io/eclipse-kuksa/kuksa-



# CAN – Where is my data in a vehicle?



## **CAN DBC**

Many automotive busses and communication protocols use description of the data

- DBC for CAN, ARXML for various AUTOSAR communication needs, DDS-IDL for DDS systems etc.
- We focus on DBC for CAN



Can we describe the relation to VSS signals?



# **DBC overlay: Describe VSS-DBC relation in VSS**



# Variants?



# **KUKSA CAN provider**



VSS provider syncs of the vehicle with VSS model of the server

- data-provider makes sure that the actual state of a vehicle is represented in VSS (historically known as "feeder")
- actuation-provider ensure that the target value of a VSS actuator is reflected by the actual state of a vehicle





# **Running CAN provider**

● ● ■ testdemo — docker run -itrm 2024-04-15 14:50:43,975 INF0 dbcfee 2024-04-15 14:50:43,975 INF0 dbcfee 2024-04-15 14:50:43,975 INF0 dbcfee 2024-04-15 14:50:43,975 INF0 dbcfee 2024-04-15 14:50:43,976 INF0 dbcfee 2024-04-15 14:50:44,166 INF0 dbcfee 2024-04-15 14:50:44,166 INF0 dbcfee ile /data/myextvsswithdbc.json 2024-04-15 14:50:44,166 INF0 dbcfee dbc_default_values.json 2024-04-15 14:50:44,167 INF0 dbcfee 7.0.0.1:55555 2024-04-15 14:50:44,167 INF0 dbcfee Databroker must run without authen 2024(0.15 14:50:44,167 INF0 dbcfee	net=host -v ~/Documents/Dev/testdemo:/data ghcr.io/eclipse-kuksa/kuksa-can der: Reading configuration from file: config/dbc_feeder.ini der: DBC2VAL mode is: True der: VAL2DBC mode is: False der: Path to token information not given der: Starting CAN feeder derlib.dbcparser: Reading definitions from DBC file Model3CAN.dbc derlib.dbc2vssmapper: Reading CAN<->VSS mapping definitions from f derlib.dbc2vssmapper: Reading default CAN signal values from file derlib.databrokerclientwrapper: No token path specified. KUKSA.val tication!	<pre>"metadata": {     "data_type": "UNSPECIFIED",     "entry_type": "UNSPECIFIED"     } }, "fields": [     "VALUE"     ] } [ {     "entry": {         "path": "Vehicle.Chassis.Axle.Row2.Wheel.Left.Brake.Temperature",         "yalue": {             "yalue": 29.0,             "timestamp": "2024-04-15T14:51:46.574605+00:00"             },             "metadata": {             "data_type": "UNSPECIFIED",             "             "data_type": "UNSPECIFIED",             "             "</pre>
<pre>secure connection! 2024-04-15 14:50:44,167 INFO kuksa_ 2024-04-15 14:50:44,167 INFO kuksa_ 2024-04-15 14:50:44,172 INFO dbcfee 2024-04-15 14:50:44,172 INFO dbcfee can_mask': 2047}, {'can_id': 599, ' 2024-04-15 14:50:44,172 INFO dbcfee g file /data/candemo.log 2024-04-15 14:50:44,173 INFO dbcfee d to: ChannelConnectivity.REAE</pre>	client.grpc: No Noot of present, it will not be possible to use a client.grpc: Establishing insecure channel der: Setting up reception of CAN signals der: Using DBC reader derlib.canreader: Using CAN frame ID whitelist=[{'can_id': 1022, ' can_mask': 2047}] derlib.canplayer: Starting repeated replay of CAN messages from lo derlib.databrokerclientwrapper: Connectivity to data broker change	<pre>"entry_type": "UNSPECIFIED"</pre>
2024-04-15 14:50:44,173 INFO c 2024-04-15 14:50:44,173 INFO c	Hel	pful commands
2024-04-15 14:50:44,174 INFO c 2024-04-15 14:50:44,174 INFO c 2024-04-15 14:50:44,174 INFO c 2024-04-15 14:50:44,175 INFO c ft.Brake.Temperature is alreac 2024-04-15 14:50:44,176 INFO c ght.Brake.Temperature is alrea 2024-04-15 14:50:44,176 INFO c ght.Brake.Temperature is alrea 2024-04-15 14:50:44,177 INFO c	cker run -itrmnet=host -v \$(pw ovider/can-provider:0.4mapping /d ata/candemo.log	d):/data ghcr.io/eclipse-kuksa/kuksa-can- ata/myextvsswithdbc.jsondumpfile



😑 🌒 🚞 scs2rng — docker run -it --rm --net=host ghcr.io/eclipse-kuksa/kuksa-python-sdk/kuksa-client:latest — 86..

# Making an App

	Good to know
<pre>async def present(): while True: print("         Processed: " + str(nbr_vss_signals) + " VSS signals!") print("         FL brake: " + str(vss_values['Vehicle.Chassis.Axle.Row1.Wheel.Left.Brake.Temp print("         FR brake: " + str(vss_values['Vehicle.Chassis.Axle.Row1.Wheel.Right.Brake.Temp print("         RL brake: " + str(vss_values['Vehicle.Chassis.Axle.Row2.Wheel.Left.Brake.Temp print("         RR brake: " + str(vss_values['Vehicle.Chassis.Axle.Row2.Wheel.Left.Brake.Temp print("         RR brake: " + str(vss_values['Vehicle.Chassis.Axle.Row2.Wheel.Left.Brake.Temp print("         RR brake: " + str(vss_values['Vehicle.Chassis.Axle.Row2.Wheel.Right.Brake.Temp print("         Max speed since last time: " + str(vss_values['Vehicle.Speed'])) reset_vss_values() await_asyncio_sleep(5)</pre>	<ul> <li>KUKSA Python SDK is an easy way to use the KUKSA GRPC API</li> <li>You can use GRPC directly in any supported language</li> <li>You can use the KUKSA Android SDK for Smartphones, Tablets and Android Automotive</li> </ul>

ntu	command	C
UU	Commanu	5

pip install kuksa-client python app.py



# Make a Web App/PWA





# Happy End

Would have been even easier if the signal had already been there



🔸 🌒 🐨 🖌 🖕 🗟 💼 🔒 github.com/COVESA/vehicle_signal_specification/compar	те <b>Фа</b> С 🕘 🖞 + Г
COVESA / vehicle_signal_specification	Q   + • O 🗈 🗠 🚳
↔ Code ⊙ Issues 53 🕅 Pull requests 7 🗔 Discussions ⊙ Actions	s 🗄 Projects 🕮 Wiki 🛛 \cdots
Open a pull request Create a new pull request by comparing changes across two branches. If you need to, you o about diff comparisons here.	can also compare across forks. Learn more
<ul> <li>base repository: COVESA/vehicle_signal_speci ▼ base: master ▼</li> <li>bead repository: erikbosch/vehicle_signal_spe ▼ compare: erik_eye ▼</li> <li>✓ Able to merge. These branches can be automatically merged.</li> </ul>	
Add a title Brake temperatures	Reviewers ট্রে No reviews—at least 1 approving review is
Add a description       Write     Preview     H     B     I     I I I I I I I I I I I I I I I I I I I	required.  Assignees ည်ဒို No one— <u>assign yourself</u>
Brake temperatures might be a worthwhile addition to the VSS standard catalogue. This is our proposal. What do you think?	Labels වේ None yet
	Projects 💱 None yet
	Milestone 🛱



#### **Summary**

#### We have seen

- How to use the vss-tools to convert VSS into other useful formats
- How to add custom VSS signals and converting them using the tools
- How to model and add custom VSS metadata VSS signals and converting them using the tools
- How to run the Eclipse KUKSA software to work with VSS signals in-vehicle
- How to use a CAN provider automatically converting raw CAN signals into valid VSS signals using a configuration
- How to write an application using VSS signals
- How to create a webpage visualizing VSS data





25 September 2024 Copyright © 2023 COVESA

Ccelerat	DVESA the future of connected vehicles	
COVESA VSS	Vehicle Specification https://covesa.github.io/vehicle_signal_specification/	1986 VIRCH
/me	http://sdv.expert	
KUKSA	https://eclipse.github.io/kuksa.website/	
Examples	https://wiki.covesa.global/	
ETAS OSS	etvs <u>https://www.etas.com/en/open-source-software.php</u>	