VSS Composable Layers

Gunnar Andersson, Development Lead, GENIVI Alliance

GENIVI Alliance | December 5, 2019

GENIVI®

Introduction

- Vehicle Signal Specification (VSS) is finally coming together as an increasingly widely accepted way to describe vehicle signals and vehicle data in general
- The VSS file format is:
  - Fully hierarchical, making it suited for modeling many future data domains
  - Simple, text-based, **human-readable** *and* **machine-processable**
    - Supporting preferred development methods (git-repo, diff/merge)
  - Open-licensed
- The VSS ecosystem further includes and promotes common tooling and methods for transformation, code-generation and binding to existing technologies
- BUT! There are many aspects of data exchange → we need layered formats to extend the metadata of the VSS-based data taxonomy.

Copyright © GENIVI Alliance 2019

Introduction

- But! There are <u>many</u> aspects of data exchange in the future vehicle-data driven architecture.

- → we need *multiple layers* to extend the VSS-based data taxonomy.

Copyright © GENIVI Alliance 2019

VSS – Data Categories

Data Categories are needed to control both technical behavior and correct handling of differing needs, laws, and conventions around data. Examples:

- **Sensitive and non-sensitive**
- **Safety-critical**
- **Personally identifiable information**
- **Accessibility**
- **Precision / quality / reliability** – many qualitative attributes

- Laws govern the correct handling of Personally Identifiable Information
- Access control lists are required to limit access to only what is allowed
- When preparing a data exchange between system parts, a formal specification can be useful for which data is required to be exchanged between parts

Copyright © GENIVI Alliance 2019

Challenges

- Complexity of data-oriented systems increase further with the addition of data categories and access-limits.
- **Technical differences in variants** (e.g. car models have lots in common, but might redefine certain details, including the availability or use of some data)
- Diverse **markets** have **different** expectations
- **Legislation** (e.g. what is considered privacy sensitive) **differs** among countries and even amonng counties and states (e.g. differing state laws in USA)

This increasing amount of different considerations require formats that manage this complexity. We need continued use of **computer-processable** formats that are also **easy to read/**write and understand by human designers that write the file content

Copyright © GENIVI Alliance 2019

Composability

- As noted, the creation of system variants from a common basis, and variations in using those systems on different markets will require some of the defined metadata to be flexibly changed where others stay stable.
- True system flexibility comes when these separate definition files can be reused in future and possibly new combinations:
    - → we need *layers* to be *composable*!

Copyright © GENIVI Alliance 2019

Composable data model layers – details

- Because of the geographical differences and other variations, a single data definition file should not encompass all metadata
- It is required to *layer* these different aspects on top of each other, so that layers can be changed as needed
  - Different market – change the affected layer
    Updated laws – change that layer, etc.
  - Definitions (i.e. an instance of the VSS) that do not change does not need to be re-versioned because of only certain extended characteristics having changed
- We propose therefore:

- **Composable VSS Layers**

Copyright © GENIVI Alliance 2019

VSS Layers – principles

- Reuse basic VSS data format (**YAML**) and design → **Familiarity**
- Related nodes are tied simply by using the same identifier.
  - → A signal node path and name may appear in more than one file, extending its metadata.
  - Example: The main VSS (vspec) file defines **Vehicle.Chassis.Headlight.Main.Wattage**, it is per the VSS vspec format required to define description, node type, data type, etc.
  - Another layer may also reference **Vehicle.Chassis.Headlight.Main.Wattage** and add new metadata. Because its *identifier* is identical, this shall extend the original object from the .vspec file.
- Wildcards (*) can be used to match identifiers to make widely applicable definitions with minimum redundancy
- The concept allows composing (layering) any number of VSS layers
- Layers typically **add** new **metadata**, but can also **override** the value for existing metadata.

Copyright © GENIVI Alliance 2019

VSS Layers – defining the full (meta)model :

- 1) VSS layers can define new concepts and meta-concepts, which are in turn referred by concrete layers
- The meta-concept such as “Category” is instantiated, for example to “DataSensitivityLevel”, and “SecurityAccessLevel”
- 2) Where **.vspec** is the primary VSS specification file suffix, each new layer type shall use another .suffix
- (As explained before, the rest of the name/directory structure shall be identical with the corresponding .vspec, since per VSS rules the directory hierarchy and file name is part of the *node path* which is a unique identifier of the node).
- The concept **DataSensitivityLevel** is further defined by specifying what values it may have. E.g. an enum (**DataLevelPublic, DataLevelSensitive, DataLevelPrivate**), or a numerical value.
- 3) Finally, at the lowest level, a VSS extension layer can add the “metadata” values of previously defined concepts:
- Example content: **Chassis.privspec** to specify the DataSensitivityLevel of each VSS node / signal
  -
```
Headlights.Left.Powerdraw:
      - DataSensitivityLevel: DataLevelPublic
```

More examples, details

- (please help out)
- Define categories here by...
  - Grouping or Filtering
  - Lists or patterns

Concrete examples...

Copyright © GENIVI Alliance 2019

RECAP

- VSS Layers are **extensible**. Meta-concepts can be defined, each with their own file type
- The file format is always analogous with the main VSS (.VSPEC) file format (YAML, hierarchical, same node identifier)
- VSS Layers are **composable**, any number of layers are feasible
- VSS Layers take care of the issue of **separating often changing concerns** (markets, local legislation) from **stable and common parts** into separate files so that only some need to vary
- The ability to add any new metadata supports this separation
- **Adding** new **metadata** supports future ideas, and *where needed* proprietary extension
- **Overriding meta-data** also enables inheritance-like modeling patterns for various use

Copyright © GENIVI Alliance 2019

Additional design challenges

- Privacy sensitive data is not always defined per data item because each data item might not be considered sensitive, wheres the combination of a few such items might be!
- Discuss how the layered metadata approach could define a data format that takes this into account.

Copyright © GENIVI Alliance 2019

BACKUP SLIDES