

## OpenDRIVE Quality Assurance Framework

Kick-Off Workshop ASAM OpenDRIVE, Oct 09, 2018

**Audi** Electronics Venture

# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**  
**Challenges**

**02.**  
**Use Cases**  
**Requirements**

**03.**  
**Architecture**

**04.**  
**Examples**

**05.**  
**Next Steps**

# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**

**Challenges**

**02.**

**Use Cases**

**Requirements**

**03.**

**Architecture**

**04.**

**Examples**

**05.**

**Next Steps**

## Using OpenDRIVE

### Well known challenges

- › OpenDRIVE processing components do not support all alternative feature-description possibilities
  - › simulation software
  - › database editors
  - › reference sensor models
  - › visualization
  - › ...
  
- › People are not able to specify all requirements regarding format details
  
  
- › New suppliers underestimate the complexity

# Using OpenDRIVE

## New challenges

- › Demand of generating track variations for automated tests
  
- › Simulation gets more and more standard in series-development  
→ OpenDRIVE users are not the simulation experts!
  
- ➔ Stop all activities regarding database generation in I/AEV-25  
( for the moment ;- )
  
- › Current objective: Make OpenDRIVE fully operational for series-development

# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**

**Challenges**

**02.**

**Use Cases**

**Requirements**

**03.**

**Architecture**

**04.**

**Examples**

**05.**

**Next Steps**

# OpenDRIVE Quality Assurance Framework

## Use Cases

- › Is a track usable in my simulation environment?  
→ Run a whole suite of checks
  
- › Does a track fulfill my project-specific requirements?  
→ Run individual checks or add them to the test suite
  
- › Is my scenario runnable on a specific track?

Think about parameter variations in static and dynamic parts of a logical scenario!

# OpenDRIVE Quality Assurance Framework

## High-level requirements

- › Be able to
  - › integrate in to local and server based tool chains
  - › implement checks in any programming language
  - › static file checks and dynamic tests (run a simulation and observe behavior)
  - › parameterization of checks
  - › check other simulation database levels (e.g. graphical representation)
  - › traceability of issues to source code and visualization



# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**

**Challenges**

**02.**

**Use Cases**

**Requirements**

**03.**

**Architecture**

**04.**

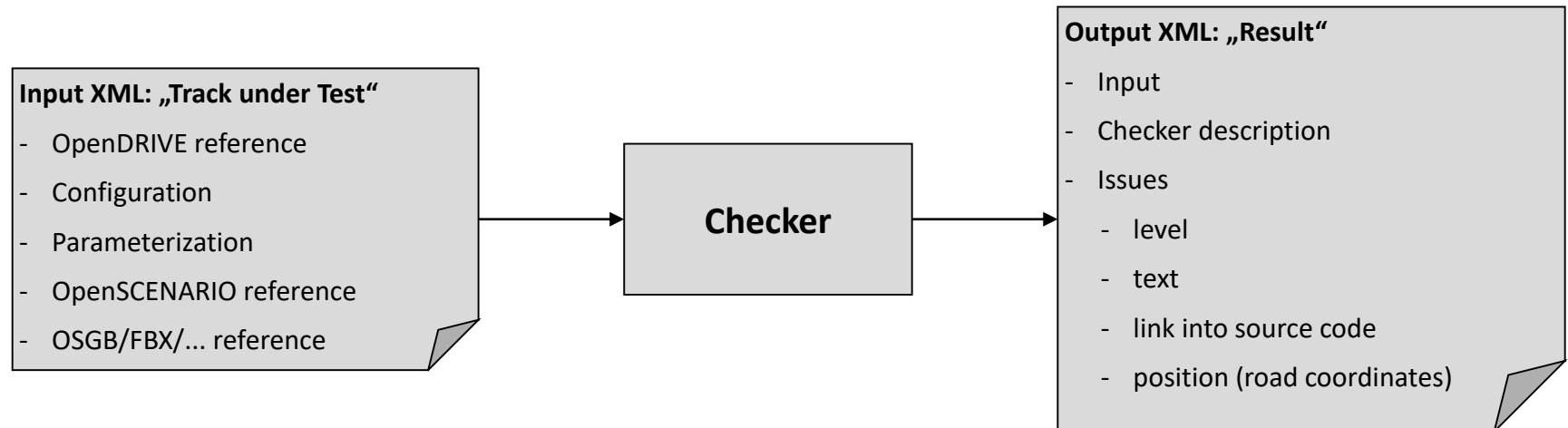
**Examples**

**05.**

**Next Steps**

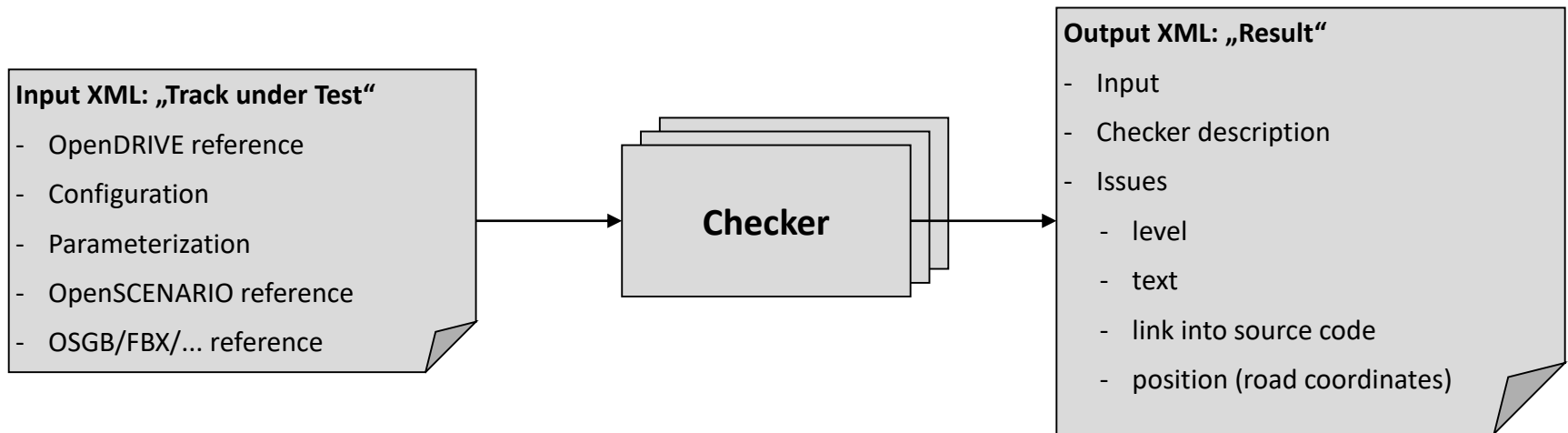
# OpenDRIVE Quality Assurance Framework

## Input and output data for a check



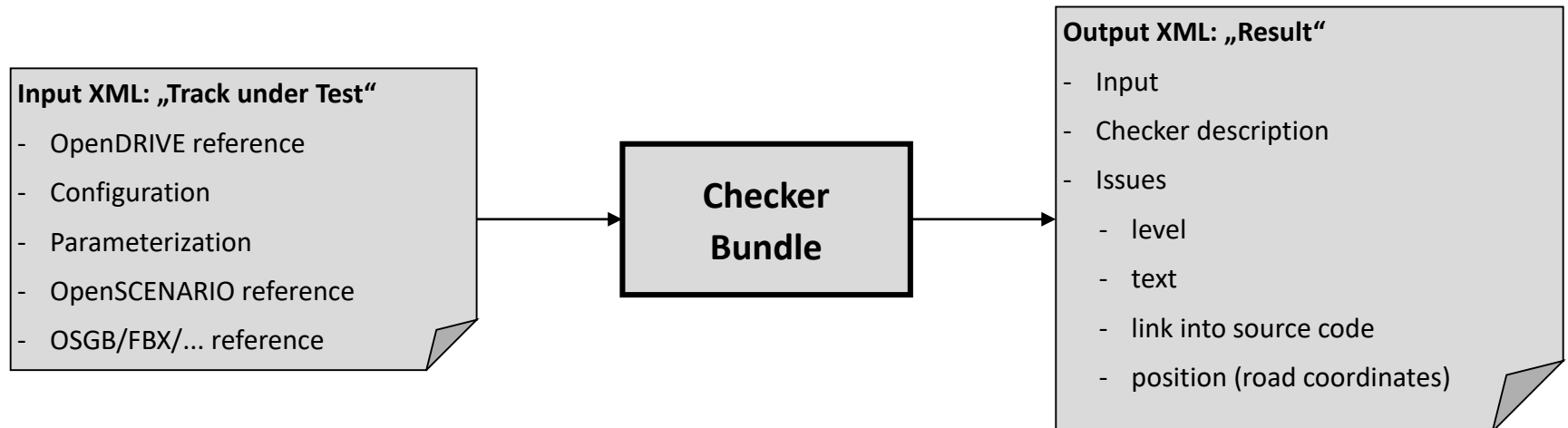
# OpenDRIVE Quality Assurance Framework

## Input and output data for a check



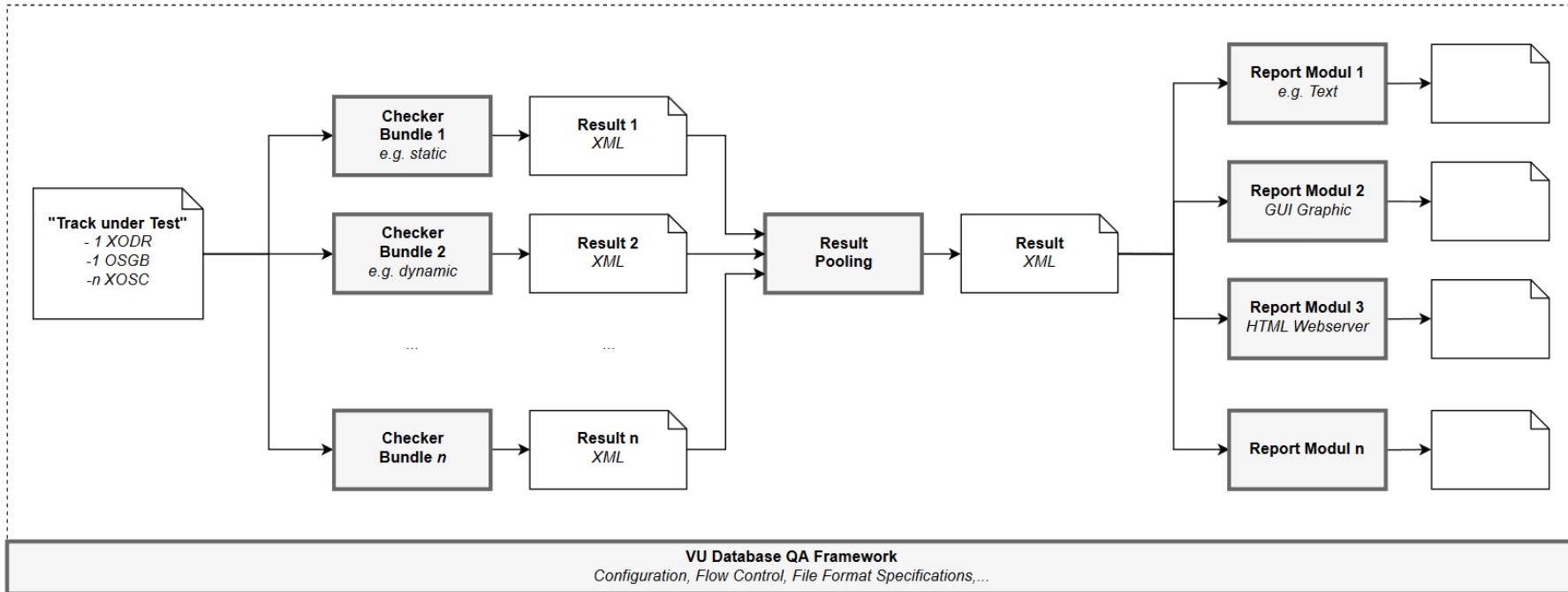
# OpenDRIVE Quality Assurance Framework

## Input and output data for a check



# OpenDRIVE Quality Assurance Framework

## Architecture overview



# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**

**Challenges**

**02.**

**Use Cases**

**Requirements**

**03.**

**Architecture**

**04.**

**Examples**

**05.**

**Next Steps**

# OpenDRIVE Quality Assurance Framework

## Output examples from text based ReportModule

CheckerBundle: XODRSchemaValidator

Description: Checks validity of the xml schema of a given xodr file.

Summary: Found 145 issues

Checker: xsdSchemaChecker

Description: Checks the xsd validity of an xodr. Found xodr version: 1.3

Summary: Found 145 issues

Error: #1503: Line:20775 Message: empty content is not valid for ...

Error: #1504: Line:20794 Message: element 'road' is not allowed for ...

(...)

# OpenDRIVE Quality Assurance Framework

## Output examples from text based ReportModule

CheckerBundle: `StatisticXodrChecker`

Description: `Statistic check of a XODR`

Summary: `Extracted 71 different data entries in 14 checkers from file`

Checker: `georeferenceChecker`

Description: `Checks the availability of the geo reference`

Summary: `Found 1 issue`

Warning: `#27: No georeference available`

Checker: `roadTypeInfoInformation`

Description: `Information about the different road types of a scene`

Summary: `Found 5 issues`

Info: `#29: lowSpeed = 4273 (overall length = 112048.35)`

Info: `#30: motorway = 16 (overall length = 15925.00)`

Info: `#31: roadTypeCount = 4`

Info: `#32: rural = 767 (overall length = 108065.65)`

Info: `#33: town = 738 (overall length = 19614.70)`



# OpenDRIVE Quality Assurance Framework

## Output examples from text based ReportModule

CheckerBundle: StaticXodrChecker

Description: Checks the consistency and structure of the virtual track.

Summary: 180 issues found, 2 elevationChecker issues found, ...

Checker: plainViewChecker

Description: Verify start with end headings: Delta between heading angle...

Summary: 175 issues found

Warning: #3: RoadId=12216 | s=1.86478 | delta of heading in transition...

(...)

# OpenDRIVE Quality Assurance Framework

## Example from GUI based ReportModule (in development)

The screenshot displays the ReportModule\_GUI interface, which is divided into several sections:

- CheckerBundle list:** Located at the top left, it lists available checker bundles: Select All, SchemaXodrChecker, StaticXodrChecker, and StaticXodrChecker.
- Checker list:** Located in the middle left, it lists specific checkers: xsdSchemaChecker, elevationChecker, lateralProfileChecker, plainViewChecker, laneSectionChecker, laneOffsetChecker, roadCounter, junctionCounter, signalCounter, and neonInfoInformation.
- Issue list:** Located at the bottom left, it displays a list of error messages, such as "Row:2614 Column:130 Message: attribute 's\_offset' is not declared for element 'line'".
- Issue details:** Located at the bottom right, it provides detailed information for a selected issue, including the checker bundle (SchemaXodrChecker), description (Checks validity of the xml schema of a given xodr file.), and summary (Found 31 issues).
- Source code:** Located on the right side, it shows the XML source code for the OpenDRIVE file, with a specific line highlighted in yellow.

The source code snippet shown is:

```
</link>
<width sOffset="0" a="3.2707220776796726" b="2.8308909316442782e-05" c="-0.0013635267889491975" d="6.57635914.
<width sOffset="19.253360993173601" a="3.2351778070544048" b="0.020657455810444363" c="-0.0002884338651992386
<width sOffset="22.055459079231252" a="3.2912708176853158" b="-0.013220945163064414" c="0.0059778109397535225
<roadMark sOffset="0" width="0.13532986770408481" weight="standard">
<type name="broken">
<line length="2.89652876185012" space="2.9626618854473925" width="0.13532986770408481" s_offset="6.31288
</type>
</roadMark>
<roadMark sOffset="22.055459079231252" type="solid" width="0.13532986770408481" weight="standard" />
</lane>
<lane id="-2" type="driving" level="false">
<link>
<predecessor id="2" />
<successor id="2" />
</link>
<width sOffset="0" a="3.6007323026585354" b="0.028985859115115972" c="-0.007637386735036688" d="0.0004814043
<width sOffset="8.4867060689718485" a="3.5911704832966977" b="0.0034335319596352021" c="-0.000597692459456322
<width sOffset="22.055459079231252" a="3.508649488437154" b="0.015766716836334533" c="-0.0069937183675152044"
<roadMark sOffset="0" type="none" width="0" />
</lane>
</right>
</laneSection>
</lanes>
<objects>
<object id="268437191" type="poles" s="1.1745853566999211" t="-10.188921350196143" zOffset="-1.0583452161997684" ro
<object id="1354" type="" name="SgRArrowLeft.fl.t" orientation="none" validLength="0" s="6.643139105358717" t="-1.6
</objects>
<signals>
<signal dynamic="no" id="1689" type="274" subtype="58" country="DEU" zOffset="2.9416547838002316" s="1.174585356699
<signal dynamic="no" id="1735" type="276" subtype="1" country="DEU" zOffset="2.192125203214573" s="1.1745853566999
</signals>
</road>
<road name="" length="4.7045256286463539" id="2101308" junction="2097235">
<link>
<predecessor elementType="road" elementId="1051" contactPoint="start" />
<successor elementType="road" elementId="1266" contactPoint="end" />
</link>
<type s="0" type="town" />
<planView>
<geometry s="0" x="-593.38341635885286" y="487.18097613565624" hdg="1.3231444029532076" length="4.7045256286463539"
<paramPoly3 aU="0" bU="4.7016105598688007" cU="0.0043727027981068671" dU="0.0014597994320420036" aV="0" bV="2.2
</geometry>
</planView>
<elevationProfile>
<elevation s="0" a="424.35127563896089" b="0.0024115543013287079" c="0.0025741156256117604" d="-0.00018238577267757
</elevationProfile>
```

# OpenDRIVE Quality Assurance Framework

## Example from issue visualization (planned)



- › Visualization engine generates 3D model from OpenDRIVE
- › Remote controlled by the GUI

# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**  
**Challenges**

**02.**  
**Use Cases**  
**Requirements**

**03.**  
**Architecture**

**04.**  
**Examples**

**05.**  
**Next Steps**

# OpenDRIVE Quality Assurance Framework

## Next steps

- › Finish proof-of-concept implementation to fulfill all high-level requirements
  - meta-model for the development of checker bundles
  - ready-to-use reporting modules (text based, GUI and visualization)
  
- › Define general requirements for OpenDRIVE files and implement the checks
  
- › Release the framework, common checks and reporting modules as a product

**Should be done in the Open Source Community and/or in the ASAM context!**

# OpenDRIVE Quality Assurance Framework

## Agenda

**01.**  
**Challenges**

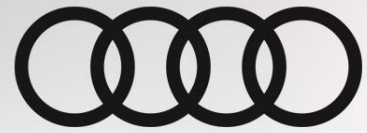
**02.**  
**Use Cases**  
**Requirements**

**03.**  
**Architecture**

**04.**  
**Examples**

**05.**  
**Next Steps**

**06.**  
**Questions / Comments?**



**Audi** Electronics Venture

