Boosting the Development of OpenDRIVE through Integration Into Standardised GIS Frameworks

ASAM International Conference, 2019-12-11, Dresden

Michael Scholz
German Aerospace Center (DLR)

Institute of Transportation Systems
German Aerospace Center

Research institutes
- Aeronautics
- Space
- Energy
- Transportation
- Security
- Digitalisation

Space administration
Project management agency
Institute of Transportation Systems

Key facts
- In Berlin and Braunschweig
- Around 200 Employees

Research fields
- Automotive
- Railway systems
- Traffic management
- Multi-modal and public transport

Area of work
- Fundamental research
- Conception and strategy development
- Prototyping
Our research infrastructure ...
... and our Testbed of Lower Saxony
Road networks with OpenDRIVE
Fancy driving simulations ...
... require highly-detailed road network data
... require highly-detailed road network data
OpenDRIVE?
OpenDRIVE: open industry standard

- XML-based database
- Hierarchical structure
OpenDRIVE: open industry standard

- XML-based database
- Hierarchical structure
- Detailed lane modelling

see OpenDRIVE Format Specification, Rev. 1.5
OpenDRIVE: open industry standard

- XML-based database
- Hierarchical structure
- Detailed lane modelling
OpenDRIVE: open industry standard

- XML-based database
- Hierarchical structure
- Detailed lane modelling
- Elements refer to an imaginary reference line

see OpenDRIVE Format Specification, Rev. 1.5
OpenDRIVE: open industry standard

- XML-based database
- Hierarchical structure
- Detailed lane modelling
- Elements refer to an imaginary reference line
- Road topography (3D) und topology
  - continuous geometry definition

see OpenDRIVE Format Specification, Rev. 1.5
OpenDRIVE: reference line geometry representation

```
| | speed
| | planView
| | geometry
| | line
| | spiral
| | arc
| | poly3
| | paramPoly3
| | elevationProfile
```

OpenDRIVE Format Specification, Rev. 1.4
OpenDRIVE: discrete geometry anchor points

```xml
<planView>
  <geometry s="0.0" x="604944.1037" y="5792860.1272" hdg="3.5148" length="9.7589">
    <arc curvature="9.0884E-4"/>
  </geometry>
  <geometry s="9.7589" x="604935.03" y="5792856.5285" hdg="3.5237" length="12.0">
    <line/>
  </geometry>
</planView>
```
OpenDRIVE: continuous geometry evolution

<planView>
  <geometry s="0.0" x="604944.1037"
            y="5792860.1272"
            hdg="3.5148"
            length="9.7589">
    <arc curvature="9.0884E-4"/>
  </geometry>
  <geometry s="9.7589" x="604935.03"
            y="5792856.5285"
            hdg="3.5237"
            length="12.0">
    <line/>
  </geometry>
</planView>
Application-based discretization (sampling)

„Everyone is doing this!“
Application-based discretization
OGC Simple Feature primitives

Simple Feature Access - Part 1: Common Architecture

1.2.1 OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture
   Document #: 06-103r4
   Type: IS

1.2.0 OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture
   Document #: 06-103r3
   Type: D-IS

1.1.0 OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture
   Document #: 05-126
   Type: D-IS

OGC® Standards
- 3D Tiles
- 3dP
- ARML2.0
- Cat: ebrIM App Profile: Earth Observation Products
- Catalogue Service
- CDB
- CityGML
- Coordinate Transformation
- Filter Encoding
- GML in JPEG 2000
- GeoAPI
- GeoPackage
- GeoSciML
- GeoSPARQL
- Geography Markup Language
- GeoRSS
- Geospatial eXtensible Access Control Markup Language (GeoXACML)
- Geospatial User Feedback (GUF)

DLR.de • Chart 22 > ASAM International Conference > Boosting OpenDRIVE through GIS > Michael Scholz > 2019-12-11
OGC Simple Feature primitives

POINT($x_1 \ y_1$)

LINESTRING($x_1 \ y_1$, $x_2 \ y_2$, ..., $x_n \ y_n$)
Fitting OpenDRIVE into the Simple Feature model

```xml
<planView>
  <geometry s="0.0" x="604944.1037" y="5792860.1272"
           hdg="3.5148"
           length="9.7589">
  </geometry>
  <arc curvature="9.0884E-4"/>
  <geometry s="9.7589" x="604935.03"
            y="5792856.5285"
            hdg="3.5237"
            length="12.0">
  </geometry>
  <line/>
</geometry>
</planView>
```

```
LineString(
  604944.1037 5792860.1272,
  604752.81 5792819.10,
...)
```

```
LineString(
  604935.03 5792856.5285,
  604754.39 5792810.73,
...)
```
Live demo

QGIS with OGC Simple Features and WMS/WFS
Conclusion: “Don’t re-invent the wheel”

• Geometry discretization could/should be based on OGC Simple Features

• Benefit from well-established tools in GIS domain:
  • Free/open frameworks for Java, C++, Python, … and web development
  • Super-easy ad hoc combination with arbitrary geo-data
  • Direct conversion into 100+ other formats: KML, GML, GeoJSON, CSV, Shapefile, SQLite, XLSX, … → GDAL: “One library to rule them all”
  • Standardized web services already available (OGC WMS, WFS, …)

• ASAM OpenDRIVE Area Concept Project introduces GIS aspects into OpenDRIVE
https://youtu.be/diEnIUT6HmA