

# Improved Data Usability: From an analysis of the GML Data Specifications to Alternate Encodings

Thorsten Reitz, wetransform GmbH

# Agenda

- Study for GDI-DE in 2017
  - How can we improve the default encoding to make INSPIRE Data more usable?
- Work for MIG 2017.2 on Alternate Encodings in 2018/2019
  - How can we use alternate encodings to make INSPIRE Data more usable?
  - What is the current state of data usability for GML and GeoJSON in the most important GIS tools?

# Analysis of the INSPIRE GML Encoding

July to August 2018

# The GDI-DE Analysis

## **Context:**

- INSPIRE Fitness for Purpose discussions
- Timeframe: 4 weeks (April 2017)

## **Objectives:**

- Determine concrete issues by use case with the current data specifications
  - Interoperability issues
  - Implementation issues
- Quantify issues and their impact where possible
- Suggest improvements, with a focus on the Technical Guidance

# Methodology of the Analysis

Create a Factual basis  
for recommendations



- BU-core\*d
- sd
- ef
- us-\*
- nz

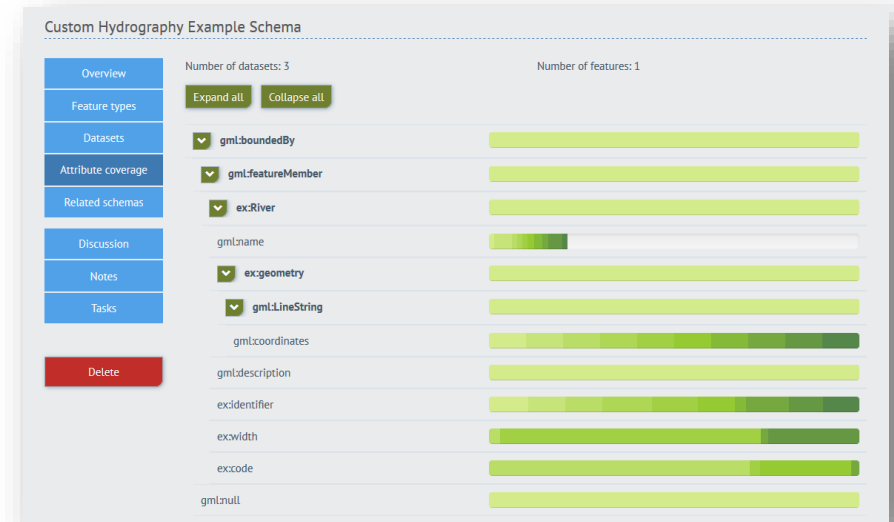
# Use Cases

- Data Management (what problems does this pattern solve for data management in a relational database?)
- Data Exchange
- Data Portrayal
- Data Harmonization
- Data Analysis in a Desktop GIS
- Data Publishing through View Services
- Data Publishing through Download Services

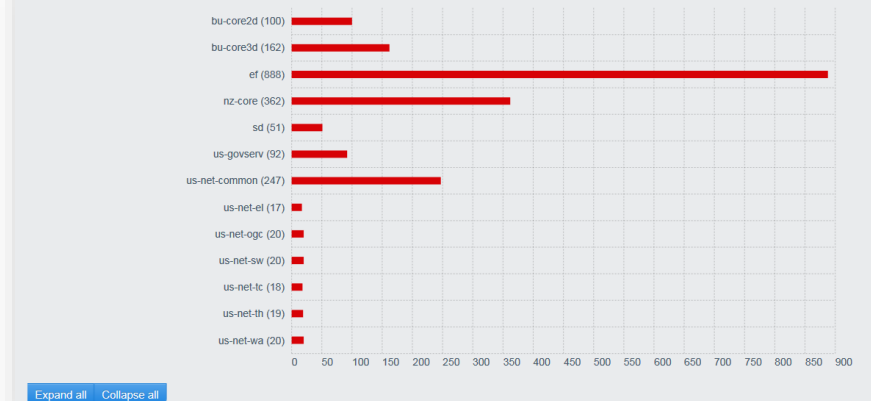
# Schema and Data Analysis

- element depth
- use of base types
- reference types
- use of choices and substitutions constructs
- statistics of property type and occurrences
- statistics of frequently used concepts like Voidable, UnitOfMeasure
- statistics on geometry properties usage
- Schema coverage through known data sets

... and more would be possible.

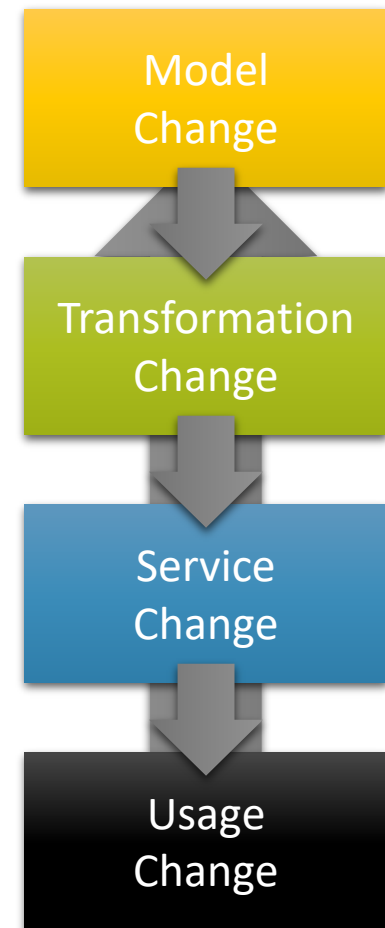


12. Number of complex properties used in all schemas



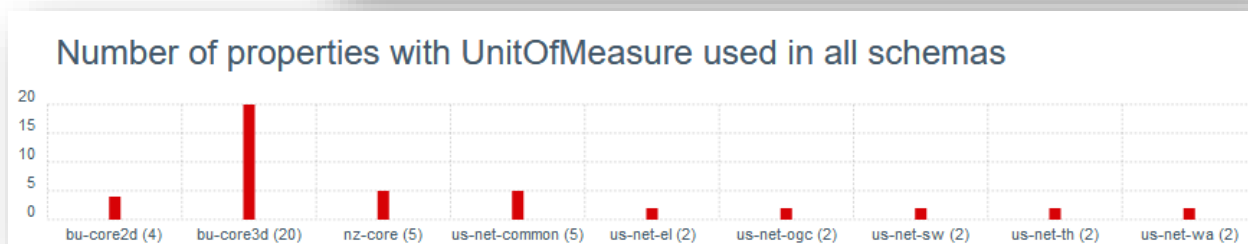
# Tools used: Agile Standardisation in hale connect

- Model-driven
  - No Schema Language Mismatches between steps
- Data-driven
  - Inform every step with real-world data
- Usage-driven
  - Instantly validate data fitness-for-purpose
- Collaboration
  - Versioning, Forking
  - Comments, Tasks, Notes





# Schema Analysis: Example



# Identified Problems by Use Case

## Data Management

- OO vs. Layers
- OO vs. RDBMS/ODBMS

## Data Exchange

- Resolution of file-external references
- Support for dataset fragments

## Data Portrayal

- Nested properties
- Code list references
- Complex geometry model (Building)
- leastDetailedViewingResolution

## Data Harmonisation

- Missing Codelists
- Semantic mismatches (classification)
- Semantic ambiguity
- Networks/Topologies

## Data Analysis

- Complex attribute structures
- Multiple geometries per „layer“

## Data Publishing

- GetFeatureInfo and complex schemas
- Some geometries and constructs not supported

# (Some) Recommendations

- Flatten some structures where cardinality is usually 1
- Simplify attributes representing measures and remove obligation to provide `UnitOfMeasure` for attributes that are `Voidable`
- Simplify `xs:choice` elements in data models
- Reduce use of substitution groups
- Avoid features with multiple geometry attributes
- Provide middleware to support alternative encodings
- Provide code list references in XML Application Schemas
- Provide alternative data models for View Services
- **Use alternative logical models for specific elements**
- **Use alternative/additional encodings**

# Conclusions

- Medium to high degree of complexity in INSPIRE schemas comes from multiple sources:
  - The structure reflects the thematic structure of the real-world features behind.
  - The specifications reuse existing standards.
  - The data models are object-oriented and were created using an MDA approach.
- Comparison to other national or international standards shows that overall complexity is in line with expectations
- XML schema is still the only established mainstream technology for a formal definition of data exchange formats
  - Significant tooling available

# Easy INSPIRE Extensions

- Touch-friendly View
  - Based on UML Model
- Recommendation Engine
- Easy re-use of components of other schemas/models
- Supports subset of UML/XSD concepts
  - Classes, Choices, Enumerations
  - Interhittance, Aggregation, Reference

The screenshot displays the 'hale CONNECT' web application interface. At the top, there is a navigation bar with 'Data', 'Themes', 'Schemas', and 'Transformation Projects' tabs. A progress indicator shows six steps: 1. Select creation method, 2. Add information, 3. Add Type, 4. Create relationship (current step), 5. Define properties, and 6. Confirm creation. A warning message asks to 'Decide how BorderControlStation should be related to types of other schemas'. Below this, a UML diagram shows 'BorderControlStation' with relationships to 'Building' (via a solid line with an open arrowhead), 'Aggregation' (via a dashed line with an open diamond), 'Association' (via a dashed line with an open arrowhead), and 'Composition' (via a solid line with a filled diamond). A green button labeled 'I need a suggestion' is positioned below the diagram. At the bottom, there are 'Back' and 'Continue' buttons. The footer contains copyright information for wetransform GmbH 2017, a link to 'Documentation', and the 'we transform' logo.

# Work on Alternate Encodings (GeoJSON)

November 2018 to June 2019

# Outline for an Alternate Encoding

- Scope
  - Use Cases
  - INSPIRE Themes
  - Cross-cutting INSPIRE requirements
- General Encoding Rules
- Conformance Classes
- Mapping to the Default encoding
- ATS/ETS, Examples
  
- Optional: Model Transformations (for Simplification, Flattening)

# How can the encoding be used?

Can I use \_\_\_\_\_ ? Settings

x | Feature: CSS Grid Layout

## CSS Grid Layout CR

Usage % of all users

Global 85.36% + 2.93% = 88.3%

unprefixed: 85.36%

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Current aligned | Usage relative | Date relative | Apply filters | Show all | ?

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android *	Blackberry	Opera Mobile *	Chrome Android	Firefox Android	IE Mobile	UC for Android
		2-39	4-28											
		3 40-51	1 29-56		10-27									
6-9	2 12-15	4 52-53	4 57	3.1-10	1 28-43	3.2-10.2								
2 10	16-17	54-62	58-69	10.1-11.1	44-55	10.3-11.4		2.1-4.4.4	7	12-12.1			2 10	
2 11	18	63	70	12	56	12	all	67	10	46	70	63	2 11	11.8
		64-65	71-73	TP										



# What to optimize the encoding for?

- What general and/or domain specific use case(s) was the encoding in the example developed for?
- Usability in current applications
- What INSPIRE themes would it be suitable for?
- Which specific technical problems does this encoding solve?
- Which INSPIRE general encoding and data specification requirements does it address?
  - Should the encoding be applicable to all themes, including those that use 3D geometries, coverages/TINs, Topologies, ...?
- What clients or other software was this encoding tested with? What were the results?

# Questions? Feedback?

+49 6151 155 408

[info@wetransform.to](mailto:info@wetransform.to)

[www.wetransform.to](http://www.wetransform.to)

[www.linkedin.com/company/wetransform-gmbh](http://www.linkedin.com/company/wetransform-gmbh)

[https://twitter.com/tr\\_xsdi](https://twitter.com/tr_xsdi)