S-100 GML Format Way Forward

Raphael Malyankar

S-100 WG 6

10-14 January 2022

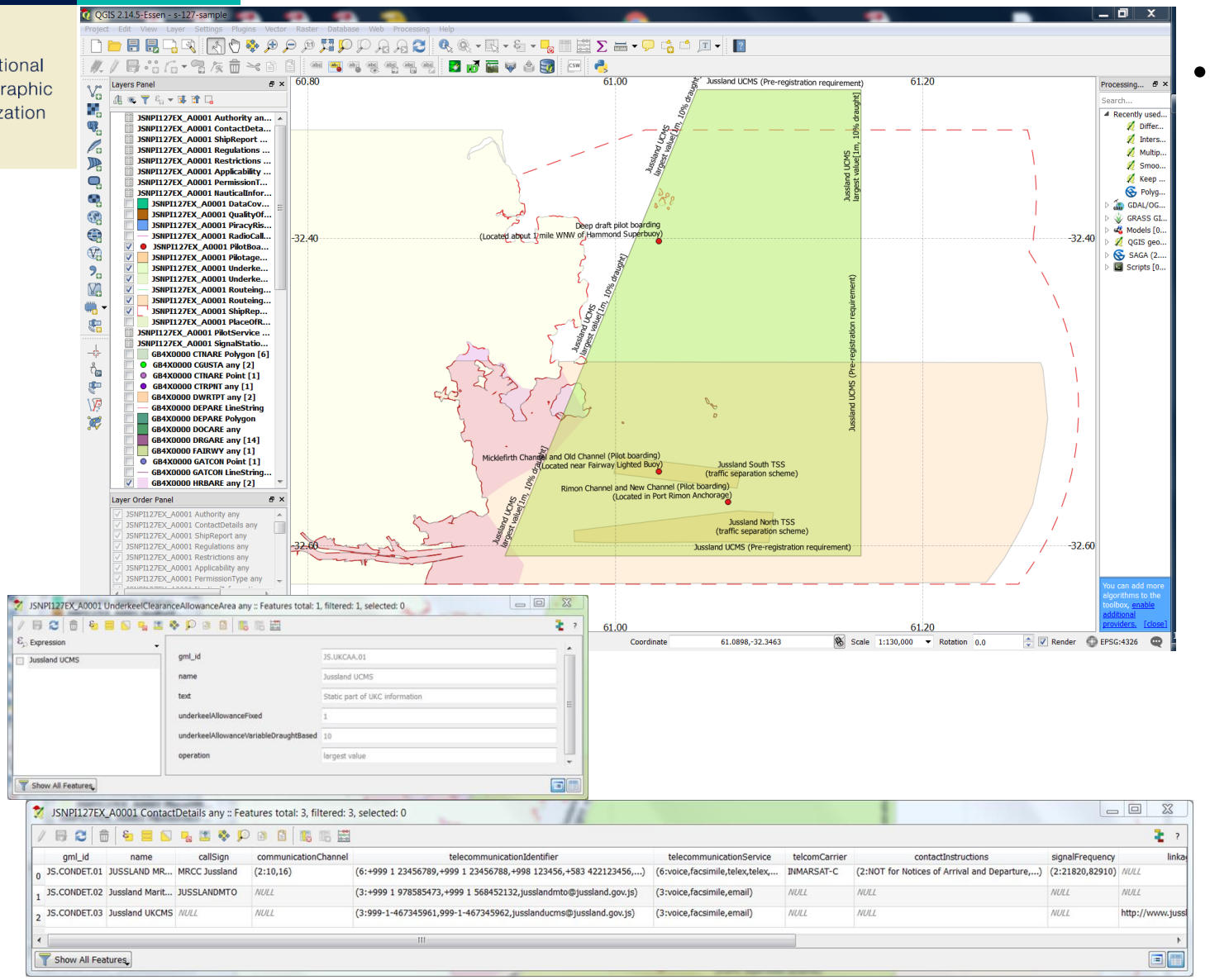
MAIN TECHNICAL IDEAS

International Hydrographic Organization

Reduce to widely used GML constructs	• Use standard GML mappings • Drop S-100 variations to ISO GFM and Geometry
Adopt best practices from S-122, S-123, S-127, S-121	• Rationalize spatial types • Drop 1/1 mapping to ISO 8211 format • Reduce number of options
Relationship to FC and Application Schema UML	• Expand description of mapping to feature catalogue
Software libraries	• Use GML-specific off-the-shelf libraries, generic XML libraries may involve a lot more work for GML
Solution strategies	• Generate code classes – if needed – from FC • Adopt generic strategy for ingest • Leverage XSLT and Schematron to reduce effort
S-100 Developer Resources	• S1OOP Wiki • Other common resources?

QGIS - S-127 SAMPLE DATASET

International Hydrographic Organization



- The QGIS data ingest strategy works with S-122, S-123, and S-127 sample datasets that conform to the Ed. 4.0 GML profile.
  - Loads both feature and information types.
  - Product-specific GML schema is not needed by QGIS – all it needs to know that it is a GML 3.2 file.
  - Can ingest all information except for S-100's special extensions to the ISO 19109 model and GML (e.g., multiple spatial attributes in a feature instance, associations, scale attributes on spatial types).

PROPOSED PLAN

International Hydrographic Organization

Update Part 10b for Ed. 5.0.0

- Firm up technical ideas to be applied to the revision
- Discuss revisions on GitHub + VTCs as needed

Update GML profile

Common resources and best practices for solutions

- Document on S1OOP Wiki
- Common resources? TBD.

Use S-131 (Marine Harbour Infrastructure) as a testing ground.

- S-131 is currently under development.
- Confirm with NIPWG.

Github discussion - 6.11 Revision of GML Annex #23: https://github.com/IHO-S100WG/TSM8/issues/23

Supplementary Slides Technical Details

The following slides are for information only

Discussion & further development of details can continue on the GitHub site mentioned on the previous slide or by VTC

USE ‘WELL-KNOWN’ GML CONSTRUCTS

- Use UML ↔ GML mappings as specified by the GML standard, e.g., *Real* ↔ *xsd:double*.
- Drop implementations of S-100 variations of the ISO 19109 GFM.
  - The GML specifications implement ISO 19109, and common GML software is written accordingly. Processing S-100’s variations requires customizations or extensions.
- Examples ... and exceptions:
  - Drop scale ranges on spatial attributes – create a copy of the feature instance instead.
    - Modeling solution exists – defining scale ranges as feature attributes.
  - Disallow multi-valued geometry on feature instances – create copies each with a single-valued location.
    - Modeling solution can be defined to aggregate such copies, or to avoid duplication of thematic attributes.
    - Multi-points still OK.
    - Different feature instances can still use different types of spatial primitives (point for one, surface for another).
  - Implement information types as derived from gml:AbstractFeature, like feature types.
  - Circle and arc – implement as standard GML types.
  - Masking – one ‘extension’ S-100 probably needs to retain – because ‘common’ GML does not include it.

SIMPLIFY & STANDARDIZE ON COMMON PRACTICE

- Rationalize spatial types
  - Simplify encoding of surfaces (e.g., gml:polygon, with interpolation none or planar? Cf. S-100 § 7-4.2.3).
  - Encode geometry inline in a feature instance, instead of a separate spatial object.
    - Solution for shared geometry TBD – perhaps inline in one feature, referenced by other features?
  - Topology implementation (Node/TopoPoint, Edge/TopoCurve, Face/TopoFace) TBD, depending on validation requirements. (But “S-100 Part 7 explicitly excludes topological primitives” - S-100 § 3-6.5.1).
- Drop the old design principle of 1/1 mapping between GML and ISO 8211 formats.
  - In the last 7 years, nobody has asked about inter-conversion.
- Associations should be named properties of features instead of generic featureAssociation & informationAssociation (retain 10b-9.5.2, drop 10b-9.5.1).
- Etc.

RELATIONSHIP TO GFM AND APPLICATION SCHEMAS

International Hydrographic Organization

- The mapping between feature catalogue and GML datasets is described in 10b-14.
  - Camel case codes of features, information types, attributes as the ‘local name’ in XML tags. *Retain.*
  - Enumeration & codelists:
    - Numeric code, label, or alias. *Disallow alias*.
    - Dataset header – *attributeEncoding* field to indicate which is used. *Define in GML profile XSD*.
  - Features, information types, simple & complex attributes in FC → simple or complex types in XSD. *Current practice becomes a formal requirement.*
    - *Feature-specific limitations on enumeration values – TBD. Could be Schematron constraints (cf. S-127 1.0.0), inline restriction types, XSLT constraints, or derived types in the XSD (the last would cause lots of type proliferation in GML application schemas).*
  - Spatial objects must be encoded inline in some feature – new requirement.
  - Feature and information associations: Role name as tag - 10b-9.5.2 only. *Remove 10b-9-5.1 (generic “featureAssociation” and “informationAssociation” tags).*

SUPPORT FOR SOFTWARE AND S-100 SOLUTION STRATEGIES

- GML-specific libraries should be used, generic XML libraries are less useful.
  - Reported difficulties with using .NET XML Schema Definition Tool (*xsd.exe*) to generate classes.
  - Consider generating classes from feature catalogue instead – if classes are needed.
  - However, compliance with the GML standard means custom software for each product specification is not necessary.
    - QGIS can load GML data that complies with S-100 Part 10b and display its feature classes as layers, like other geographic data. This works with S-122, S-123, and S-127 sample data files.
      - The only knowledge QGIS needed is that it is reading GML 3.2 data. The product-specific application schema is not needed!
      - S-100-specific constructs (e.g., multiple spatial attributes in an instance, feature and information associations, complex attributes) are ignored or simplified by QGIS, but features, information types, feature geometry, and attributes and their values are read treated like other data formats.
    - Conversion to a common format is also possible using the conventions in §10b-15.