S-100 GML Format Way Forward

Raphael Malyankar

NIPWG VTC01 2023

02 March 2023

OVERVIEW

International Hydrographic Organization

- Multiple S-100-based products under development are using the S-100 GML format, now generally 4.0 or 5.0 (S-122 1.0 and S-1231.0 were prepared before Edition 4.0 was published, and S-127 1.0 as S-100 ed. 4.0 was being developed).
- The OGC GML specification is based on the “General Feature model” described in ISO 19109, and the “Spatial schema” in ISO 19107 (& other standards…), but S-100 Parts 3 and 7 extended and restricted both the ISO GFM and spatial models.
- S-100 defines unique mechanisms for portrayal of S-100 data on ECDIS and similar systems.
- The OGC GML specification is fairly complex. S-100 uses only a subset (“profile”) of the GML specification.
- Off-the-shelf GIS software can create and ingest GML-datasets
  - The GML constructs OTS software supports is often a subset of full GML and/or written for an older edition that GML 3.2.
  - S-100’s special GFM, Spatial schema, and portrayal mechanisms complicate matters in the S-100 ecosystem.
- OGC GML was designed to be “generated” from data stores.
- Like all XML information, appropriate user-friendly tools are needed to intervene between users and the raw format if users are to create or edit GML data.
- It should be possible for S-100 applications to ingest S-100 GML without bespoke implementations.
- What (more) is needed to work within these constraints and support the creation and use of S-100-based GML data?

MAIN TECHNICAL IDEAS

International Hydrographic Organization

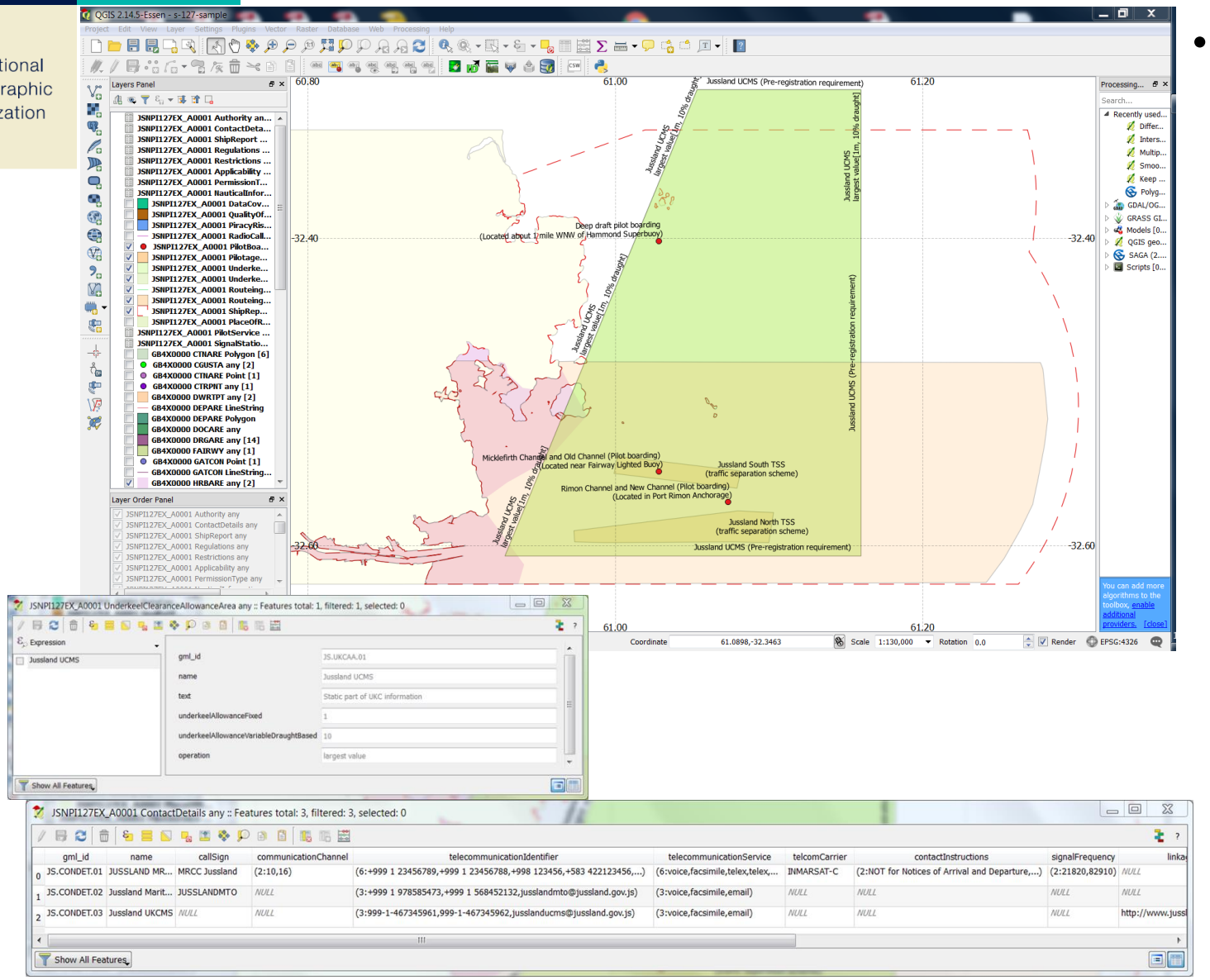
Reduce to widely used GML constructs	• Use standard GML mappings of datatypes • Drop S-100 variations to ISO GFM and Geometry for GML encodings
Adopt best practices from S-122, S-123, S-127, S-121, S-124, S-131	• Rationalize spatial types • Drop 1/1 mapping to ISO 8211 format • Reduce number of options in which curves and surfaces can be encoded

Relationship to FC and Application Schema UML	• Refine the mapping between GML schema and feature catalogue
	  • GML schema(s) are now being created from S-100 Feature Catalogues with automated processes
	  • Does the S-100 Feature Catalogue model need refinement to better support GML?

Software libraries	• Use GML-specific off-the-shelf libraries, generic XML libraries may involve a lot more work for GML
Solution strategies	• Support data development • Generate code classes – if needed – from FC • Adopt generic strategy for ingest • Leverage XSLT and Schematron to reduce effort • Improve documentation for production support
S-100 Supporting Resources	• S1OOP Wiki? • S-100 schema server https://schemas.s100dev.net • Other common resources?

QGIS - S-127 SAMPLE DATASET

International Hydrographic Organization



- The QGIS data ingest strategy works with S-122, S-123, S-127, and S-131 sample datasets[1] that conform to the S-100 4.0 / 5.0 GML profiles.
  - 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).

[1] Other products have not yet been tried yet

IHO

International Hydrographic Organization

WAY FORWARD

Test the S-100 5.0/5.1 GML Profile (Part 10b)

- Use S-131 and other S-100 5.0/5.1 conformant Products
- Gaps
- Incompatibilities with commercial and open-source OTS software
- Inefficiencies (e.g., overly complex spatial types)

Update S-100 Part 10b in S-100 Edition 6

Common resources and best practices for solutions

- Document on S1OOP Wiki
- Identify and build out common resources

As of February 2023

IHO

International Hydrographic Organization

CONCLUSION

- Use S-131 and other Ed. 5.0/5/1 –compliant products as a testbed for exploring the feasibility and interoperability of the s-100 Part 10b GML format.
- Refine and elaborate the concepts described here to make the S-100 GML encoding more compatible with general practice after more practical experience has been gained.
- Continue this topic for refinement with further experience and discussion at future NIPWG meetings.
- Ongoing input – ideas, needs, gaps, etc., etc. - is invited.

Supplementary Slides Technical Details

The following slides are for information only

Discussion & further development of details can continue on the IHO GitHub site and in NIPWG meetings

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

International Hydrographic Organization

- 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 8 years, nobody has asked about inter-conversion.
- Re-examine the encoding of associations
  - In Ed. 5.0.0 associations are named properties of features (i.e., “attributes” that reference either a feature or information type instance)
  - There are still problems with the association class encoding concept in Edition 5.0 (after Figure 10b-3). A maintenance proposal with a temporary bypass has been approved by S-100 WG for Edition 5.1. Additional follow-up is necessary.
- Other issues as they arise.

IHO

International Hydrographic Organization

Association class encoding

S-100 5.0 Clause 10b-8.4.1

10b-8.4 Associations

The profile allows associations to be encoded inline or by reference. The dataset metadata field associationEncoding shall be defined as either "reference" or "inline" to define which method is used throughout conforming datasets. In addition to the dataset metadata constraint such associations shall only be used to express composition relationships defined by the corresponding Feature Catalogue.

For bi-directional associations, the profile supports the optional encoding of the name of reverse property in the *appInfo* annotation element in the Application Schema XSD.

10b-8.4.1 Association classes

The profile allows the GML 3.3 convention for encoding of association classes using the GML 3.3 association class conversion rule, which converts association classes to an equivalent intermediate class. The figures below illustrate the conversion rule.

Where associations contain attributes in a Product Specification Feature Catalogue, this structure shall be used to realise those attributes via an intermediate Information Type. The name of the Information Type shall be defined as the name (S100_FC_Item code) of the relationship (either feature or information association) concatenated with "Type" and is not required to be defined within the Feature Catalogue.

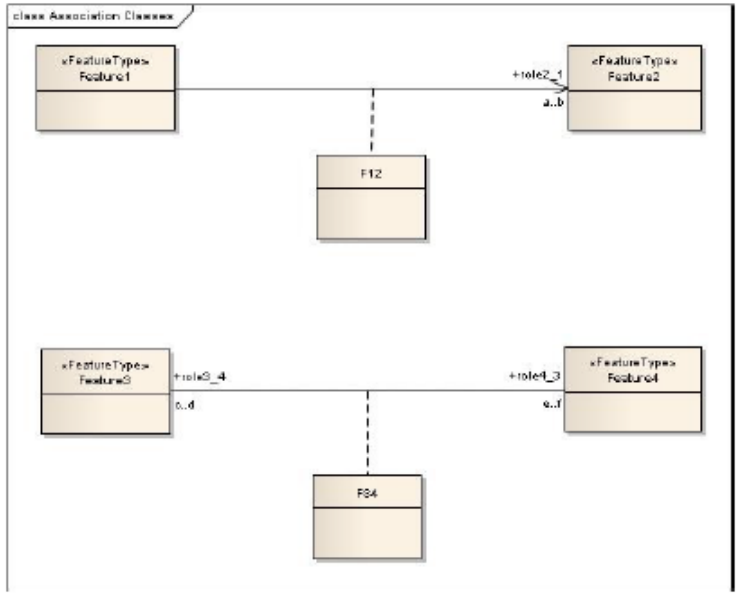


Figure 10b-2 – Model with association classes (from OGC 10-129r1 / ISO 19136-2:2015)

An example (taken from S-127) is shown in Figure 10b-3 below. Here an association class "PermissionType", corresponding to a relationship with code "Permission", is used to express the association attribute "categoryOfRelationship".

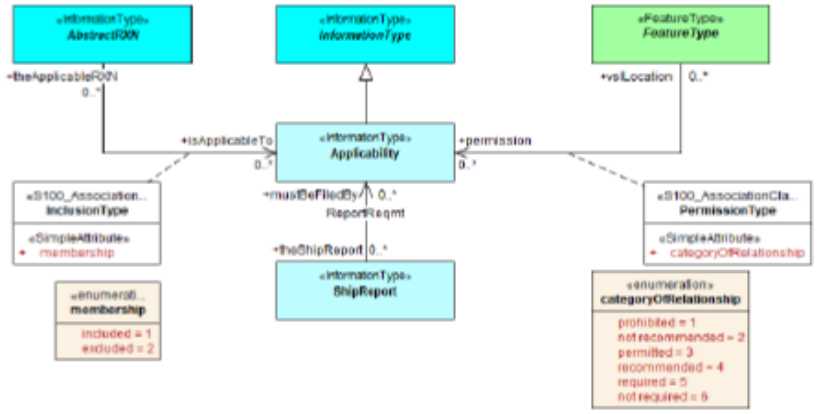


Figure 10b-3 – Example use to represent attributes of associations (S-127)

To avoid needless duplication of relationships between source/destination features and the intermediate information types, the intermediate information type shall be included inline with the referencing feature; for example:

```
<S-127:VesselTrafficServiceArea gml:id="a1">
  <categoryOfCargo>ballast</categoryOfCargo>
  <theApplicableRxN xlink:href="#R1">
    <S-127:PermissionType gml:id="res1">
      <categoryOfRelationship>required</categoryOfRelationship>
    </S-127:PermissionType>
  </theApplicableRxN>
</S-127:VesselTrafficServiceArea>
<S-127:Restrictions gml:id="R1">
  <categoryOfAuthority>coast guard</categoryOfAuthority>
</S-127:Restrictions>
```

In the Application Schema this shall be done by extension of gmlReferenceType.

```
<xs:complexType name="PermissionType">
  <xs:sequence>
    <xs:element name="categoryOfRelationship" type="categoryOfRelationshipType" minOccurs="1"
    maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>
        <xs:complexType name="theApplicableRxN">
                <xs:complexContent>
                        <xs:extension base="gml:ReferenceType">
                                <xs:sequence>
            <xs:element name="PermissionType" type="PermissionType" minOccurs="1"
            maxOccurs="1"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
```

RELATIONSHIP TO GFM AND APPLICATION SCHEMAS

- The mapping between feature catalogue and GML datasets is described in 10b-10 (Ed. 5.0.0).
  - Camel case codes of features, information types, attributes as the ‘local name’ in XML tags.
  - Enumeration & codelists
    - Numeric code and label of a “listed value” are separate types. This requires the creation software to get the combination correct (there is no intrinsic check for compatibility).
  - Features, information types, simple & complex attributes in FC → simple or complex types in XSD.
    - Feature-specific limitations on enumeration values – implemented in 5.0.0 as derived types in the XSD – leads to type proliferation in GML application schemas.
- Spatial objects can be encoded externally to all features
  - Encode spatial primitives inline and use GML references to avoid duplication?.

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-131, S-122, S-123, and S-127 sample data files. (Other GML data products have not been tried.)
      - 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 and treated like other data formats.
    - Conversion to a common format is also possible using the conventions in §10b-10.