Testbed-12 2D Test Dataset Implementation with Documentation

Table of Contents

Publication Date: 2017-06-30

Approval Date: 2017-06-29

Posted Date: 2016-11-01

Reference number of this document: OGC 16-136

Reference URL for this document: http://www.opengis.net/doc/PER/t12-A032

Category: User Guide

Editor: Guy Schumann

Title: Testbed-12 2D Test Dataset Implementation with Documentation

COPYRIGHT

IMPORTANT	Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights. Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.
NOTE	This document is a user guide created as a deliverable in the OGC Innovation Program (formerly OGC Interoperability Program) as a user guide to the work of that initiative and is not an official position of the OGC membership. There may be additional valid approaches beyond what is described in this user guide.

POINTS OF CONTACT

Name	Organization
Guy Schumann	Remote Sensing Solutions, Inc.
Stephane Fellah	Image Matters
Jeff Harrison	The Carbon Project
Andrew Smith	SSBN Ltd

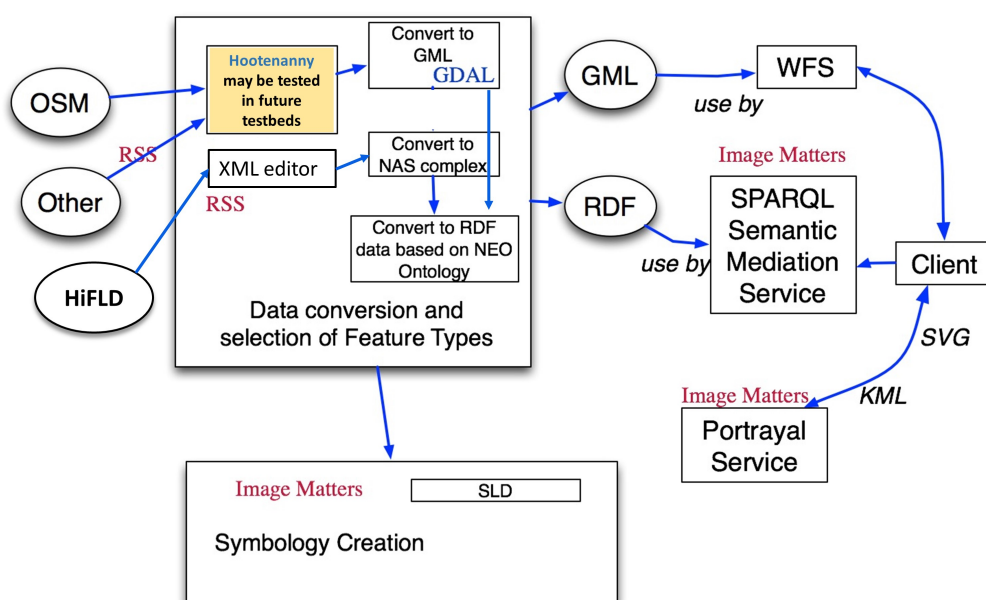# 1. Introduction

This technical report is for A032.

Figure 1. A032 workflow diagram

2. Feature Types in GML 3.2 and NAS Schema 7.2

The corrected and complete NAS V7.2 is posted at: https://portal.opengeospatial.org/files/?artifact_id=68872

Note that NAS V7.2 and the test sample data are not publicly available via the links, but were posted for internal Testbed use (outside readers will not have access to these artifacts as they are posted).

The following three test samples produced for the Testbed-12 are focusing on the SF Bay area and have been taken from the HiFLD (https://hifld-dhs-gii.opendata.arcgis.com) database:

- Geopolitical Entity (see Figure 2)

Roads

- NHD Hydrography Flowlines

The test sample for the Geopolitical Entity of CA (out of the HiFLD Geopolitical Entity data set) includes a feature-level and an attribute-level copyright and security marking and is posted at: https://portal.opengeospatial.org/files/?artifact_id=68865

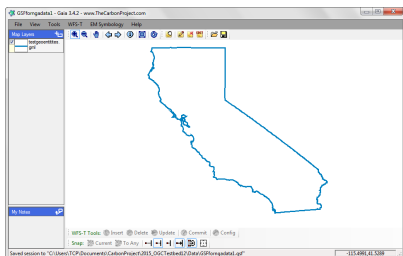*Figure 2. The Geopolitical Entity (State of CA) in GML 3.2/NAS 7.2 tested in QGIS*

3. Suggested: JSON implemention

There has been a suggestion to have a NAS JSON version at some point and this has been tested in this Testbed but it is recommended to taKe further testing and implementation of a *NAS WFS JSON* to Testbed-13. The NAS samples built during this Testbed have about 40 schemas, similar to the Geo4NIEM situation where it was necessary to build a custom WFS on CarbonCloud with a custom loader for instance. This type of complexity makes WFS implementation much more time consuming, and some operations like *Describefeaturetype* may not even validate when done.

Therefore, considering a JSON representation that is simple is suggested, also WFS REST is moving towards JSON and so is the broader community.

For more information, please refer to: https://portal.opengeospatial.org/wiki/Testbed12/Testbed12Components

Specifically: Prototype WFS 2.5 (A035) - Jeff Harrison jharrison@thecarbonproject.com

http://ows12.azurewebsites.net/wfs/featuretypes/AdministrativeSubdivision

This service focuses on deploying NGA NAS GML as JSON: NGA NAS GML as JSON output format for testing purposes only, for example to assess whether this is even feasible (it is) and what data elements must be simplified to achieve an initial deployment.

4. OSM Feature Type Datasets (Testbed-11)

Purpose: For use in demo and RDF creation

For the NEO ontology encoding and symbology styling (REASON? XXXX) and also because the NAS GML implementation is still rather complex to implement for a lot of different feature types and geometries, it was decided to use the many OpenStreetMap (OSM) feature type datasets which have been collected over the SF Bay area during Testbed-11.

Transforming the OSM datasets to GML using GDAL, the resulting datasets ended up having a number of problems that make them unusable for exploitation for the testbed, in particular for

styling. The following issues were identified:

- There was no XML Schema provided for the data, so it make it difficult for other parties to create schemas, code bindings, and validate the data.

- The tag keys and values are concatenated together using comma delimited formatting. This is less exploitable by filters and styling. This also requires a custom parser to be written.

- When concatenation is too long, the string is trimmed and followed by three dots (…)

Looking at the datasets for OSM used in Testbed-11, these are more exploitable but there are a couple of issues that needed to be fixed:

- The XSD schemas are highly redundant. They all use the same complex type definition and just differ by element name.

- The tags element contains key/value pairs, which is fine for capturing all the tags but they do not provide any name semantic.

- Data may be out of date (minor issue)

As a result, the following recommendations have been implemented:

1. Refactor the XML schema used in Testbed-11, by putting the common complex type in one schema (*OSMFeature.xsd*). See example below.

```
    <?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2016 rel. 2 sp1 (http://www.altova.com) by Stephane Fellah
(Image Matters LLC) -->
<schema xmlns:fme="http://www.safe.com/gml/fme"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.safe.com/gml/fme" elementFormDefault="qualified">
    <import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
    <element name="OSMFeature" type="fme:OSMFeatureType"
substitutionGroup="gml:AbstractFeature"/>
    <complexType name="OSMFeatureType">
        <complexContent>
            <extension base="gml:AbstractFeatureType">
                <sequence>
                    <element name="id" type="string" minOccurs="0"/>
                    <element name="timestamp" type="string" minOccurs="0"/>
                    <element name="user" type="string" minOccurs="0"/>
                    <element name="created_by" type="string" minOccurs="0"/>
                    <element name="visible" type="string" minOccurs="0"/>
                    <element name="area" type="string" minOccurs="0"/>
                    <element name="layer" type="string" minOccurs="0"/>
                    <element name="uid" type="string" minOccurs="0"/>
                    <element name="version" type="string" minOccurs="0"/>
                    <element name="changeset" type="string" minOccurs="0"/>
                    <element name="tag" minOccurs="0" maxOccurs="unbounded">
                        <complexType>
```

```
                                    <sequence>
                                        <element name="k" type="string" minOccurs="0"/>
                                        <element name="v" type="string" minOccurs="0"/>
                                    </sequence>
                                </complexType>
                            </element>
                            <element name="nd" minOccurs="0" maxOccurs="unbounded">
                                <complexType>
                                    <sequence>
                                        <element name="ref" type="string" minOccurs="0"/>
                                    </sequence>
                                </complexType>
                            </element>
                            <element name="member" minOccurs="0" maxOccurs="unbounded">
                                <complexType>
                                    <sequence>
                                        <element name="type" type="string" minOccurs="0"/>
                                        <element name="ref" type="string" minOccurs="0"/>
                                        <element name="role" type="string" minOccurs="0"/>
                                    </sequence>
                                </complexType>
                            </element>
                        </sequence>
                    </extension>
                </complexContent>
            </complexType>
        </schema>
```

2. Create element for each feature type using the common complex type OSMFeatureType if no extension is needed, otherwise create a substitution for the extended OSMFeatureType. See below for "emergency" feature type.

```xml
    <?xml version="1.0" encoding="UTF-8"?>
<schema xmlns:fme="http://www.safe.com/gml/fme"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.safe.com/gml/fme" elementFormDefault="qualified">
    <include schemaLocation="OSMFeature.xsd"/>
    <import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
    <element name="emergency" type="fme:emergencyType"
substitutionGroup="fme:OSMFeature"/>
    <complexType name="emergencyType">
        <complexContent>
            <extension base="fme:OSMFeatureType">
                <sequence>
                    <element name="emergency" minOccurs="0" type="string"/>
                    <element ref="gml:pointProperty" minOccurs="0"/>
                    <element ref="gml:multiPointProperty" minOccurs="0"/>
                    <element ref="gml:curveProperty" minOccurs="0"/>
                    <element ref="gml:multiCurveProperty" minOccurs="0"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
</schema>
```

3. Convert the OSM data within SF area using this base schema.

4. Analyze the tag keys for each feature type and create a new schema for each feature type and add a feature property on the type corresponding to each tag name used in the feature instances.

5. Reprocess the data with the new schema so it better aligns with the RDF mapping (see below).

For the RDF mapping, the OSM datasets from Testbed-11 were used to apply the methodology described above. The TagInfo Service from OSM was used to extract tag information and value information (definition in different languages, depiction). For the railroad instances, the following was implemented:

1. Parse the XML for railroads (from Testbed-11)

2. Create RDF instances of Railroad for each Feature instance in the XML

3. Add core properties to the feature instance (id, name, user, version, etc..)

4. For each tag name, create a RDF property in an ontology (OSM namespace) with the tag value to the feature instance

5. If the tag name is the same as the feature type (railway), create a SKOS Concept (except for value *Yes*) subclass of RailwayCategory and add a *dct:type* to the feature pointing to the concept.

6. Enrich the ontology by access definitions in different languages and depiction from the Tag Info Service http://taginfo.openstreetmap.org/api/4/tag/wiki_pages?key=railway

7. Enrich the schemes by access value definition from the Tag Info Service http://taginfo.openstreetmap.org/api/4/tag/wiki_pages?key=railway&value=subway

In addition, some improvements of the geometry encoding and fixes on the url encoding were made: If the feature has some point geometry the URI of the feature should be http://www.openstreetmap.org/node/{id}. If the geometry is a line it should be http://www.openstreetmap.org/way/{id}. More properties on the geometry (spatial dimension and wgs84:latitude, wgs84:longitude) and better classification using the GeoSPARQL SF ontology (Point and LineString) were also added. This should help the clients get the adequate representation of the geometry of a feature (for portrayal for example (point or polygon)).

The results of the process produces three models:

- the data model (railway instances for example).
- the ontology model (definition of properties).
- The taxonomy of category *Railway*.

Aligning the XML data to the model, provides a way to combine semantic and XML data and possibly opens the door to some powerful semantic search if the taxonomy can be improved by adding some hierarchy (narrower terms).

Using this process, XML files using the OSM datasets of Testbed-11 were created (https://portal.opengeospatial.org/files/?artifact_id=69795) for the following 11 different feature types, some containing several geometries:

- Aeroway
- Building
- Emergency
- Highway
- Landuse
- Leisure
- Military
- Power
- Public Transport
- Railway
- Waterways

Note: Both the building and highway XML files are very large, so the data include *building_ROI* and *highway_ROI* which include one sample set in case smaller file size is needed. These sets can be expanded if needed.

Feedback on how the process can be improved is important at this point in order to provide data that better fits the needs of future testbeds.

Again, note that the test sample data are not publicly available via the links, but were posted for

internal Testbed use (outside readers will not have access to these artifacts as they are posted).

5. Other Data provided for Testbed-12

General requirements for this and future testbeds: different data sets over the same area (i.e. SF Bay) which can be meaningfully combined into new products. Also, there is a need for time series data, i.e. a stack of the same data source over time to tests and demonstrate spatial/timeseries analysis.

There is a lot of raster data for SF Bay that was used during Testbed-11 (LiDAR, flood inundation simulations, etc). Most are in geotiff format but some (flooding) are available also as time-varying netcdf.

To access the data via FTP:

```
ftp.opengis.org
u:ip-data
p: EEypDat47Wuuhoo
```

5.1. Data uploaded to the OGC ftp server

There is a lot of variety including big OSM feature data & raster data from satellites, models and other sources. Description of the many data is as follows.

Data (static & time series)

Note that data are either in the latlon (WSG84, epsg 4326) projection or the NAD or WGS84 UTM Zone 10N for California.

- San_Francisco_Bay_GML_from_OSM.zip: GML (standard schema) OSM various feature data (0.5 GB) over the SF Bay area. Note that this is so people can use it in threads for now. The plan is that I (RSS) will upload some sample building XML file with the NAS 7.0 schema soon and hopefully someone on Testbed-12 can serve that via a WFS or the like.

geo_phys_data folder (satellite soil moisture, satellite rainfall and observed and simulated flooding as well as LiDAR and NED elevation data)

- Flooding_MODIS (from NASA GSFC): MODIS_Jan2_2015_raster250m.tif: This is a flood probability map from imagery aggregated over some days. MODIS_Jan2_2015_UTM10N_WGS84.shp: This is the standard flood mapping shapefile from the NASA real time flood mapping and it shows the Jan 2015 flood event in the Bay area. Very nice to include in a real demo!

- Floodzone_sealevelrise: This is a USGS project where they assume 100 cm sea level rise and it shows the land that would be permanently under water.

- GPM_imerge: This is standard NASA GPM iMerge data showing time-series geotiff rainfall intensity (in mm I assume). I covered the Bay area but resolution is 10 km pixel so please let me know if you want bigger area. Please use these data for now to test if it does what you are

hoping for in your thread.

- satellite_soilmoisture: This is standard ESA SMOS data showing time-series geotiff soil moisture (in volumetric soil moisture). I covered the Bay area but resolution is 25+ km pixel so please let me know if you want bigger area. Please use these data for now to test if it does what you are hoping for in your thread.

- LiDAR_2m_SFSU: This is a very nice top quality LiDAR DEM put together by the San Francisco State University. It covers the area used in Testbed-11 to run flood simulations (see below)

- NED_elevation: This is the standard National Elevation Dataset elevation from USGS. Very big files. I just grabbed this as they come from the NED server. It covers the entire Bay area (actually for the flooding in Testbed 11 we fused the NED with the LiDAR to allow faster runs. Here are all the USGS elevation data and the ftp info is easy to grab too in case people want more. It's an excellent service http://viewer.nationalmap.gov/basic/?basemap=b1&category=ned,nedsrc&title=3DEP%20View

Flood_simsANDsurgeH folder

- selfe_ssh_01g_storm.nc: Please consider this a "test". I would be interested in seeing whether these type of 3-D surge model output time-series point data can be served. I can deliver a much bigger area and larger time-series if desirable. Let me know. These are tidal surge heights in meters relative to zero (mean sea level) provided in hourly time stepping. The file structure is as follows with ssh denoting sea surface height:

```
'lat' 1x1 struct 84 'single'
'lon' 1x1 struct 84 'single'
'hour' 1x1 struct 840 'single'
'ssh' 1x2 struct [84,840] 'single'
```

- obs5m_maxdepths: This is a 5m 2-D hydrodynamic simulation of the SF Bay surge flood event of 1996. It couples two models –SCHISM surge model and LISFLOOD-FP for inland flood simulation (units are m depth at max peak above sea level)

- obs5m folder (please use only .wd files!): The .wd files are ascii grid files at 5m resolution at the UTM zone10 North projection which can be easily transformed into geotiff. The files show inundation depth above sea level (bare ground) in meters. There are 100 files as hourly time series, so 100 hours from the start of the surge in winter of 1996 (start: file 0000.wd). This ascii grid is an ESRI ascii grid (header info included) and is the standard output of the LISFLOOD-FP flood model (http://www.bristol.ac.uk/geography/research/hydrology/models/lisflood). Note that these simulations are only for a smaller area in the Bay but I'm producing at the moment these type of flood depths all over the Bay area and will upload these as soon as possible. Because of size, the results will be in netcdf (nc) as time-series.

Please email schumann@remotesensingsolutions.com if you have any question and thanks to OGC Testbed-11 and USGS, SFSU, NASA, ESA for most of these data.

Time-varying flood simulations in NETCDF (SF_1in100.nc)

- Together with my colleagues at SSBN Ltd (same as Testbed-11), we have generated a netcdf (nc) file that contains time-varying flood depths together with a tif for area coverage - 30 m Landsat

image - and mp4 movie for a feel. I think this is absolutely great data to test and I wish to stress that this is state-of-the-art flood modeling and at the moment we are still pretty unsure how to deal/distribute flood simulations in time-varying format to end-users etc, so a great opportunity for this testbed. Please note: this dataset should only be used as test data and only within Testbed-12 for now. If we need to have it available for future testbeds, this should be possible. File format/variable are as follows (netcdf4):

```
Dimensions:
rows = 4800
cols = 4800
timestep = 108
LatitudeExtent = 2
LongitudeExtent = 2
cellsize = 1
Variables:
rows
      Size
            4800x1
      Dimensions
            rows
      Datatype
            double
cols
      Size
            4800x1
      Dimensions
            cols
      Datatype
            double
timestep
      Size
            108x1
      Dimensions
            timestep
      Datatype
            double
LatitudeExtent
      Size
            2x1
      Dimensions
            LatitudeExtent
      Datatype
            double
LongitudeExtent
      Size
            2x1
      Dimensions
            LongitudeExtent
      Datatype
            double
```

```
cellsize
    Size
        1x1
    Dimensions
        cellsize
    Datatype
        double
depth
    Size
        4800x4800x108
    Dimensions
        rows,cols,timestep
    Datatype
        double
```

Appendix A: Revision History

Table 1. Revision History

Date	Editor	Release	Primary clauses modified	Descriptions
June 10, 2016	G. Schumann	.1	all	initial version
July 01, 2016	G. Schumann	.1	all	added "other data" description
July 15 10, 2016	G. Schumann	.1	all	added XML → RDF description from S. Fellah
September 08, 2016	G. Schumann	.1	all	added NAS JSON suggestions from J. Harrison
October 20, 2016	G. Schumann	.2	all	incorporate feedback from G. Buehler