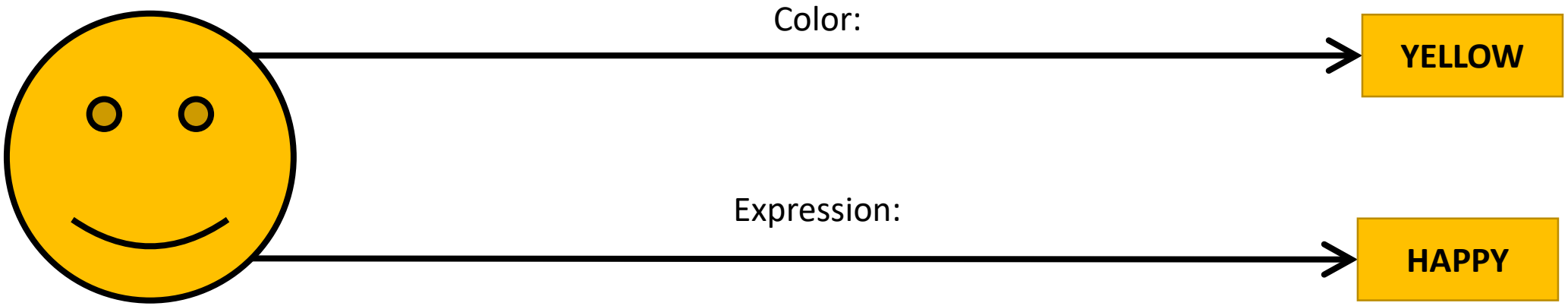
OGC APIs and SensorThings - Bridging the Gap

Kathi Schleidt, Hylke van Der Schaaf, Sylvain Grellet

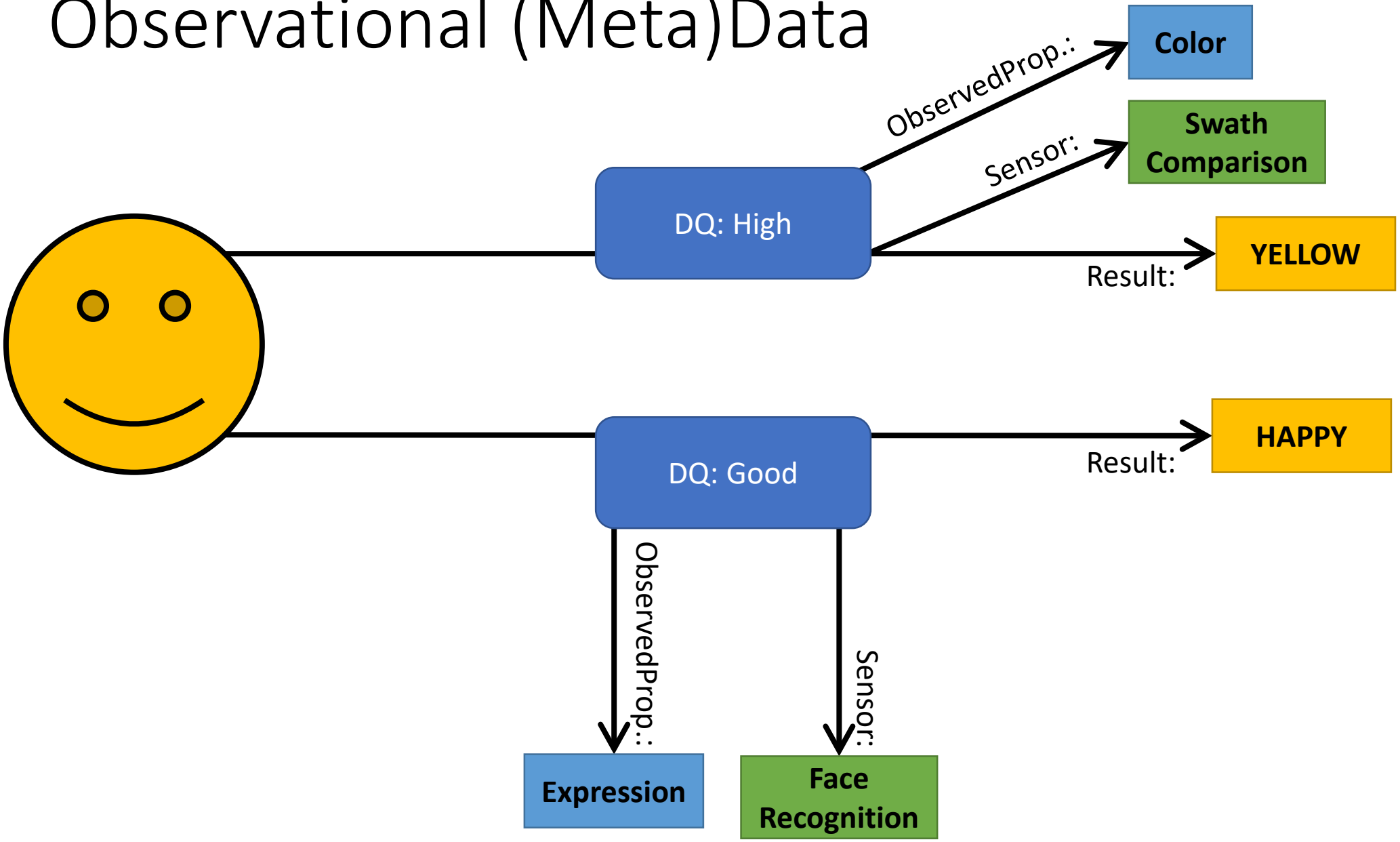
OGC APIs and SensorThings

Bridging the Gap

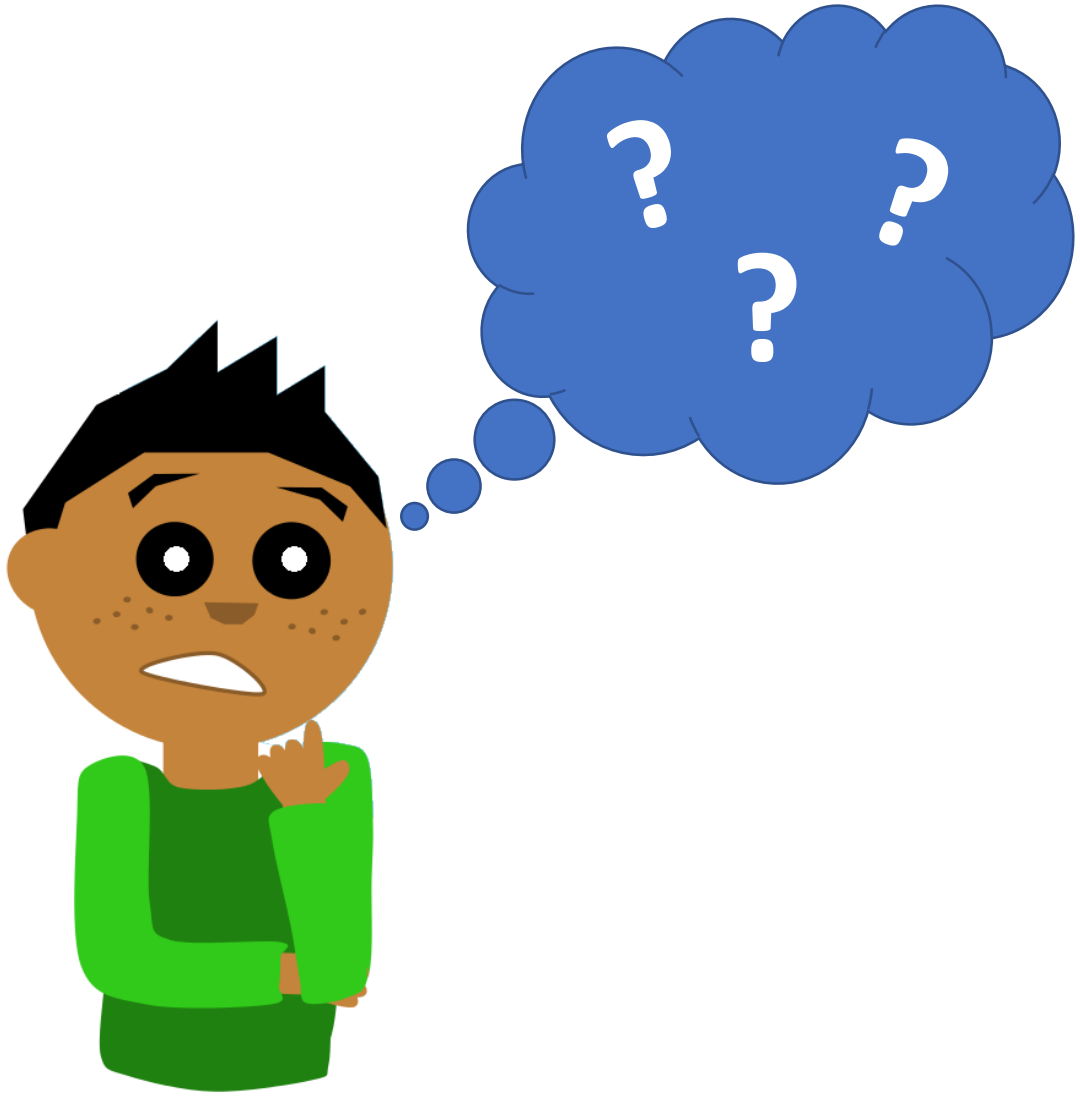
Observational (Meta)Data

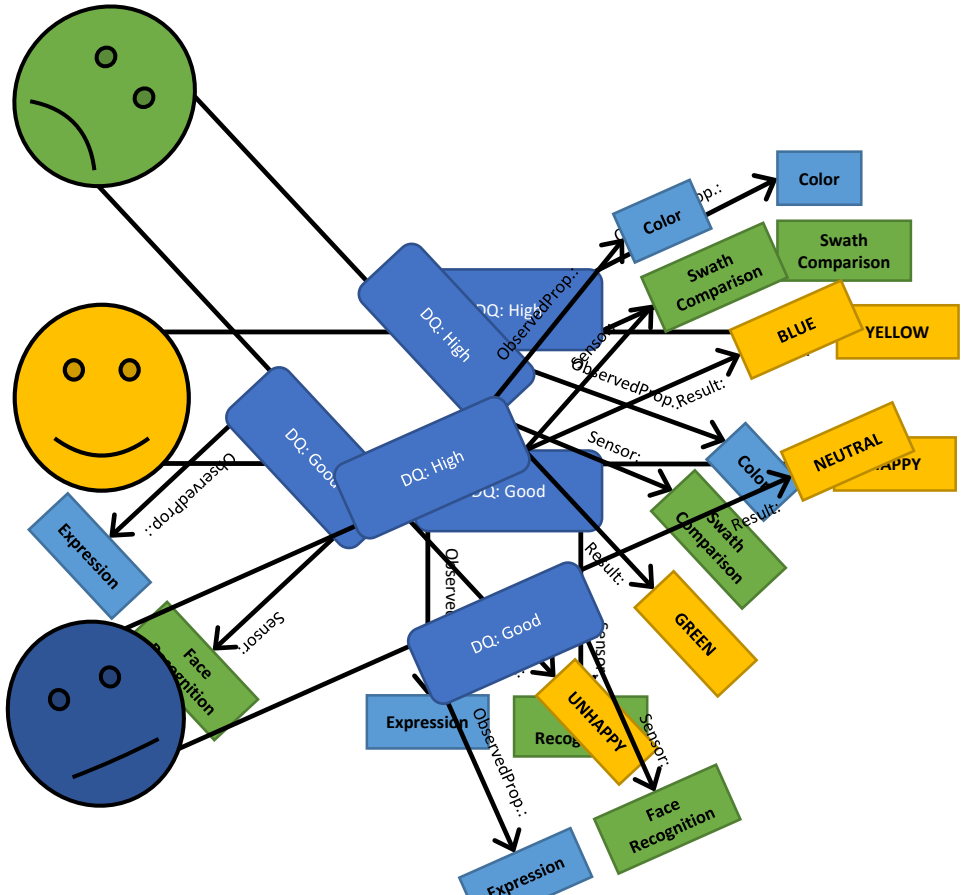


Observational (Meta)Data



Using Observational (Meta)Data





Using Observational (Meta)Data

Data Quality?

Methodology?

UoM?

Object	Descriptor	Reference
Face1	Color	Yellow
Face1	Expression	Happy
Face2	Color	Green
Face2	Expression	Unhap
Face2	Name	Bob
Face3	Color	Blue
Face3	Expression	Neutral
Face3	Name	Mary

Object	Color	Expression	Name
Face1	Yellow	Happy	Abbie?
Face2	Green	Unhappy	Bob
Face3	Blue	Neutral	Mary

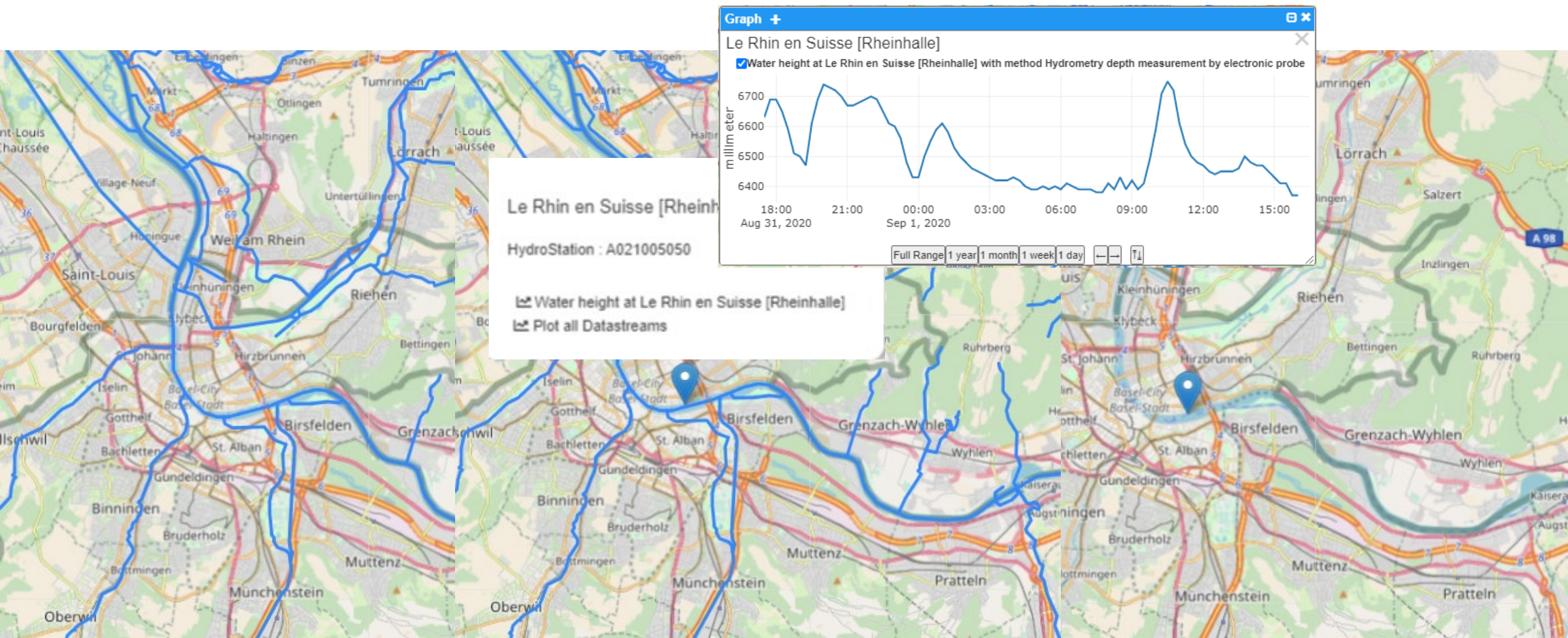
SensorThings API V1.1

A Quick SensorThings V1.1 Update

Some Powerful Pragmatic Additions

- Extended Properties: all STA Types now have a Properties element for flexible extension
- Custom ResponseFormat: implementations can provide custom responses. Some new developments in FROST:
  - GeoJSON: While native STA nests GeoJSON, this response format turns it around, nests all STA data within the GeoJSON object
  - CSV: Simple flexible rectangular formats ready to be plugged into R
- Internal linkages (FROST?): allow additional internal links between Things (or other objects), links fully available for query and access

But – how to we link to OGC API – Features?



STA vs. OGC API – Features: Issues

Overlapping Data

- STA includes both FeatureOfInterest as well as Thing within core data model
- Essential for complex queries
- Spatial objects should ALSO be available within OGC API – Features context

→ Currently investigating common landing page options, inspiration from SELFIE

STA vs. OGC API – Features: Issues

Different Identification

- https://sensorthings.brgm-rec.fr/SensorThingsGroundWater/v1.0/Things(**1**)
- https://service.datacove.eu/geoserver/ogc/features/collections/hy-n:WatercourseLink/items/_**100006785**?f=application%2Fgeo%2Bjson

STA vs. OGC API – Features: Issues

Different identifiers - both API endpoints behind a resolver

- https://iddata.eaufrance.fr/api/stapiHydrometry/Things(174)
  → redirects to a baseServerURL/FROST-Server/v1.1/$1

- https://iddata.eaufrance.fr/api/hydrometryFAPI/ef:EnvironmentalMonitoringFacility/items/A021005050?f=application%2Fgeo%2Bjson
  → redirects to a baseServerURL/ogc/features/collections/$1

Reconciliation using HTTP URIs - content negotiation or via passing f= MediaType

- https://iddata.eaufrance.fr/id/HydroStation/A021005050?f=application/json
  → SensorThings API
- https://iddata.eaufrance.fr/id/HydroStation/A021005050?f=application%2Fgeo%2Bjson
  → OGC API Feature

STA vs. OGC API – Features: Common Landing Page

OGC SELFIE inspired application/JSON


```
{
    "type": "Identifier",
    "id": "A021005050",
    "links": [{
            "title": "A021005050 via OAF",
            "type": "application/geo+json",
            "rel": "alternate",
            "profile": "OAF",
            "href": "https://iddata.eaufrance.fr/api/hydrometryFAPI
                        /ef:EnvironmentalMonitoringFacility
                        /items/A021005050?f=application%2Fgeo%2Bjson "
        }, {
            "title": "A021005050 via STA",
            "type": "application/json",
            "rel": "alternate",
            "profile": "STA",
            "href": "https://iddata.eaufrance.fr/api/stapiHydrometry/Things(174) "
        }, {
            "title": "This document",
            "type": "application/json",
            "rel": "self",
            "href": "https://iddata.eaufrance.fr/id/HydroStation/A021005050"
        }
    ]
}
```

STA vs. OGC API – Features: Common Landing Page

SELFIE defined LD+JSON payload when NO content type defined (try in PostMan)


```
{
    "@context": […],
    "@type": ["Feature", "sosa:Sample"],
    "@id": "https://iddata.eaufrance.fr/id/HydroStation/A021005050",
    "name": {…},
    "description": "Surface water hydrostation from French national flood monitoring system (SCHAPI",
    "spatialCoverage": {…},
    "label": […],
     "subjectOf": [{
                              "schema:url": "https://iddata.eaufrance.fr/id/HydroStation/A021005050",
                              "@type": "foaf:Document",
                              "primaryTopic": "ef:EnvironmentalMonitoringFacility",
                              "format": [
                                "application/gml+xml", "application/geo+json", "application/ld+json"
                              ],
                              "conformsTo": "https://inspire.ec.europa.eu/id/document/tg/ef",
                              "provider": "https://iddata.eaufrance.fr/"
                        },{
                              "schema:url": "https://iddata.eaufrance.fr/id/HydroStation/A021005050",
                              "@type": "foaf:Document",
                              "primaryTopic": "http://www.opengis.net/def/..../15-078r6:Thing",
                              "format": [
                                        "application/json"
                              ],
                              "conformsTo": "http://www.opengis.net/def/docs/15-078r6",
                              "provider":https://iddata.eaufrance.fr/}],
     "geometry": {…}
}
```


Static example

Query Logic

STA vs. OGC API – Features: Issues

Different Query Logic

- Complex queries across multiple objects core in STA
- Still unclear where OGC API – Features is going here

→ Clear is that such complex queries are essential to working with the underlying O&M data model!

STA vs. OGC API – Features: Issues

Issues in Presentation

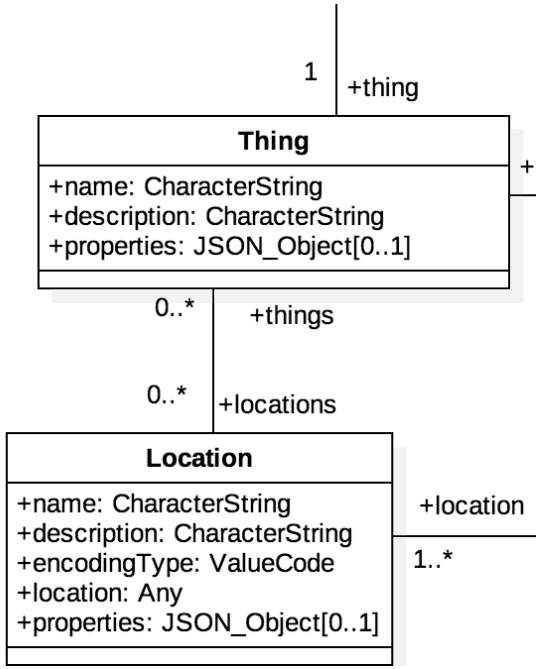
- Missing native support for integration with common web mapping tools
- Understanding of Data Model still not widespread

→Developed simple JavaScript Library for STA Maps:

https://github.com/DataCoveEU/STAM

Multi Resolution Data

- European NUTS regions with demography data
  - Regions in 5 scales. Witch to choose?
- A STA Thing can have multiple Locations (Geometries)
  - →Add All scales!
  - →Store the scale for each in the properties
  - →Build cool queries!
    (example) (example)



→https://datacoveeu.github.io/API4INSPIRE/maps/Demography.html
Powered by STAM

Conclusions

- Uptake of APIs is MUCH faster than OWS
  - Already have communities utilizing both APIs
- Use Cases require BOTH Spatial and Observational Data
  - Many different domains
  - STA being creatively used way beyond sensors (see DemographyThings)
- We NEED a way to combine both API worlds
- While we appreciate the deliberation that has gone into OGC API, we do NOT see how we can give up the power of OData

Thanks for your Attention!

Kathi Schleidt
Kathi@DataCove.eu

Hylke van den Schaaf
hylke.vanderschaaf@iosb.fraunhofer.de

Sylvain Grellet
S.Grellet@brgm.fr