GETTING TO KNOW

- OBSERVATIONS, MEASUREMENTS AND SAMPLES

- OGC SENSORTHINGS API PART 1

Sylvain Grellet – BRGM
ENSG – mastère GDM – 2023-01-20

Observations, Measurements & Samples

ISO / OGC standards

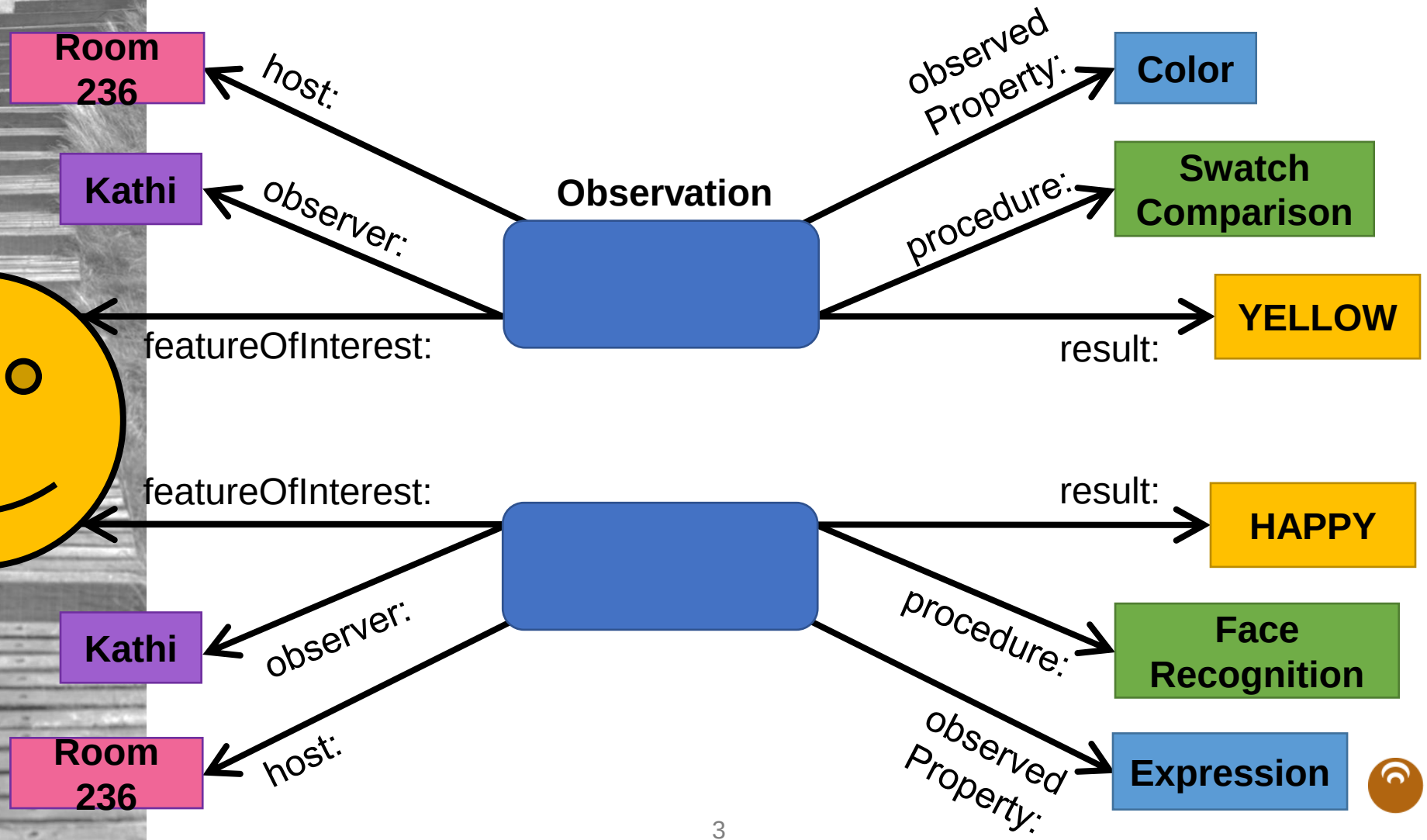
color: YELLOW

expression: HAPPY

```
"smiley": {
    "color": "Yellow",
    "expression": "Happy"
}
```

Observations, Measurements & Samples

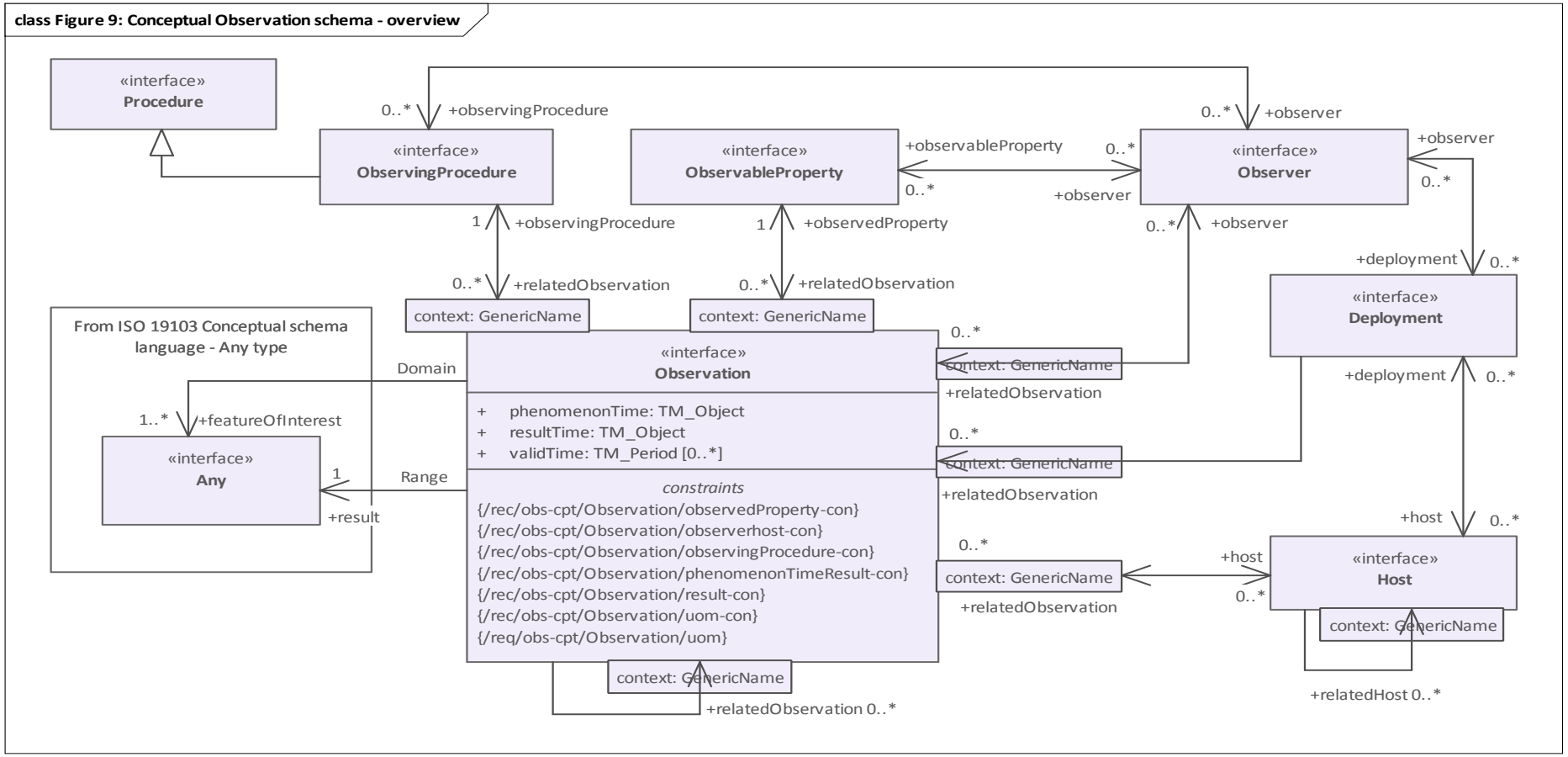
ISO / OGC standards



Géosciences pour une Terre durable
brgm

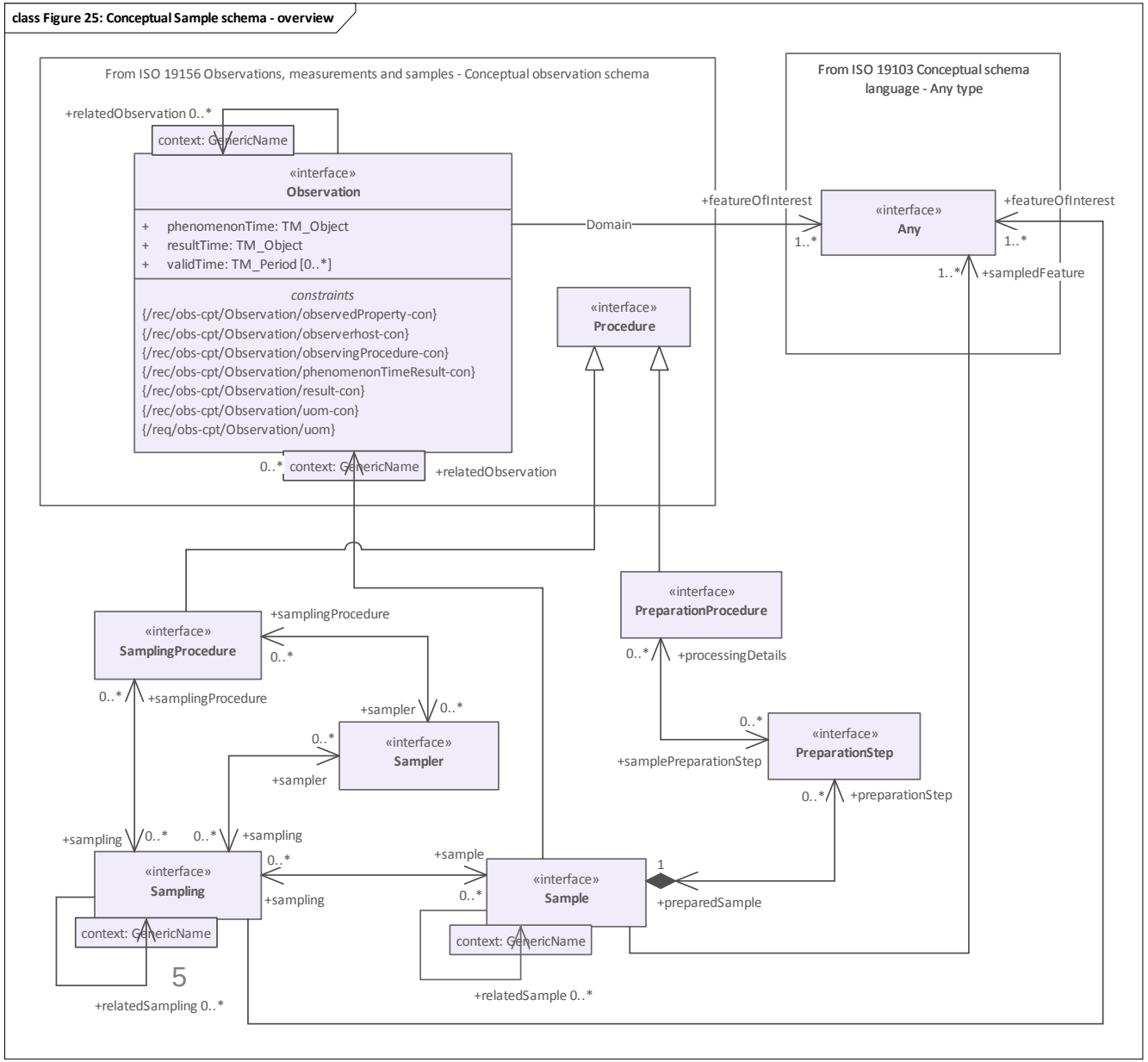
Observations, Measurements & Samples

2 core parts – Observation

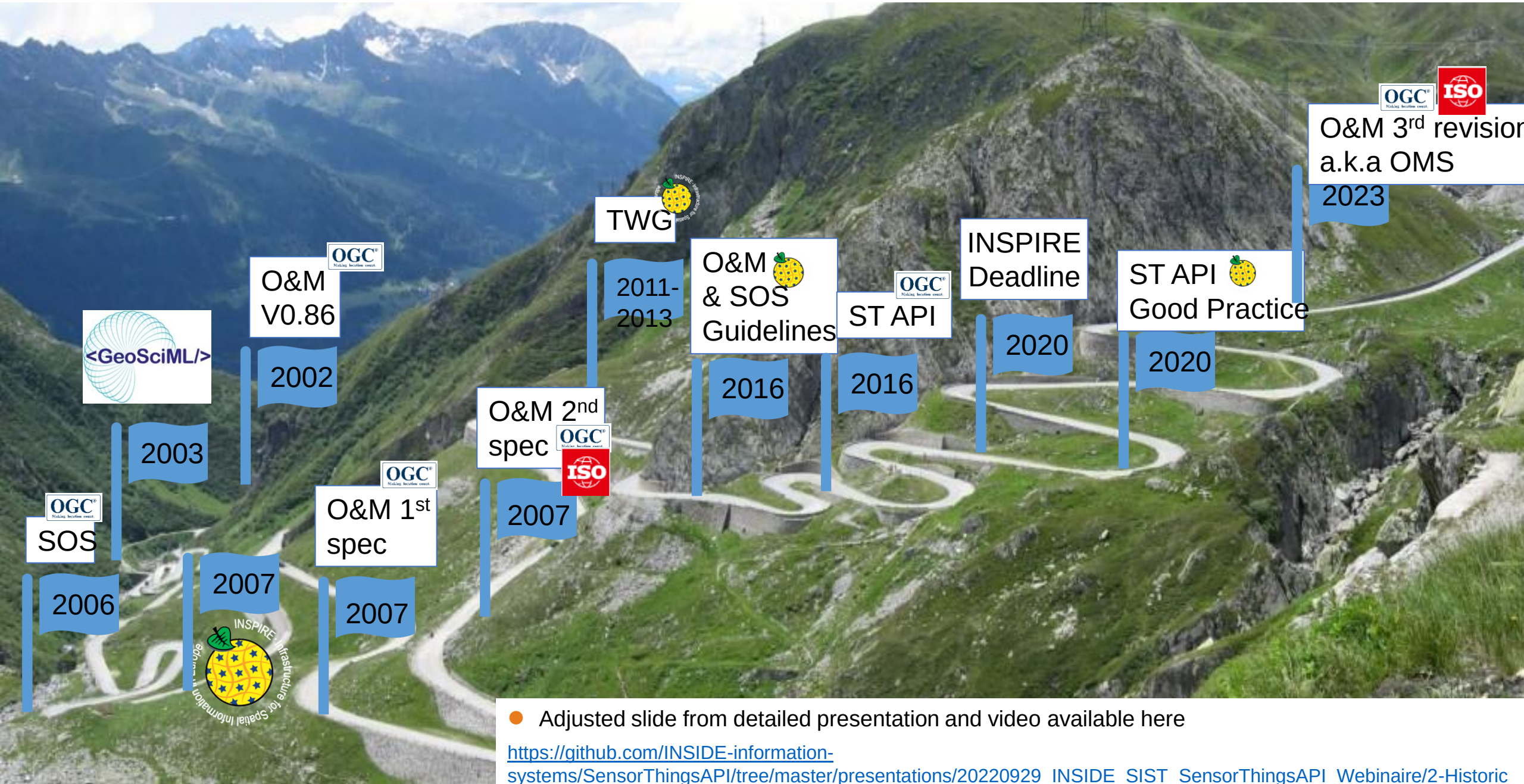


Observations, Measurements & Samples

2 core parts – Sample



The long and winding road from O&M to OMS



- Adjusted slide from detailed presentation and video available here

https://github.com/INSIDE-information-systems/SensorThingsAPI/tree/master/presentations/20220929_INSIDE_SIST_SensorThingsAPI_Webinaire/2-Historic

OGC SensorThings API Part 1

Excerpt from 3 presentations

- Available here (including videos)

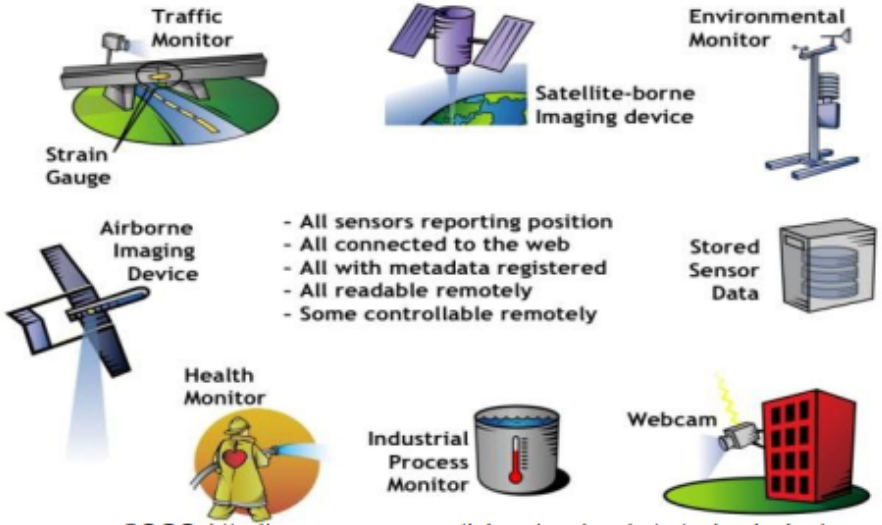
https://github.com/INSIDE-information-systems/SensorThingsAPI/tree/master/presentations/20220929_INSIDE_SIST_SensorThingsAPI_Webinaire/3-ST_API_presentation

OGC SensorThings API Part 1 - context

OGC & IoT?

- IoT deals with Sensors and Actuators
- Sensors and Actuators have Location
- OGC Sensor Web Enablement (SWE)
  - Enable developers to make all types of sensors, transducers and sensor data repositories discoverable, accessible and useable via the Web
  - Since 1990 by NASA → 2001 in OGC
  - SensorML
    Sensor Observation Service (SOS)
    Observations & Measurements (O&M)
  - Sensor Data & Metadata
  - Core of INSPIRE

The basis

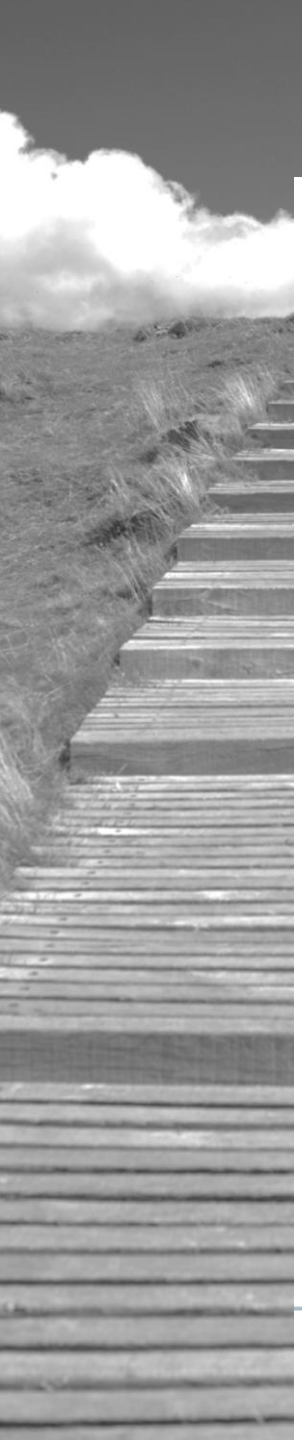


©OGC: http://www.opengeospatial.org/ogc/markets-technologies/swe

OGC SensorThings API Part 1 - context

From SWE to SensorThings

- "Old" SWE Standards
  - XML Encoded
  - SOAP bindings
  - Requires tools for use
  - Complex in use
    - No easy browsing
    - No pagination
    - No pub/sub
    - No updating
    - No delete
- Time for an update → SensorThings API



OGC SensorThings API Part 1 - context

OGC SensorThings API

The Update

- A standard for exchanging sensor data and metadata
  - Historic data & current data
  - JSON Encoded
  - RESTful
  - Adapting OASIS OData URL patterns and query options
  - Supporting ISO MQTT messaging

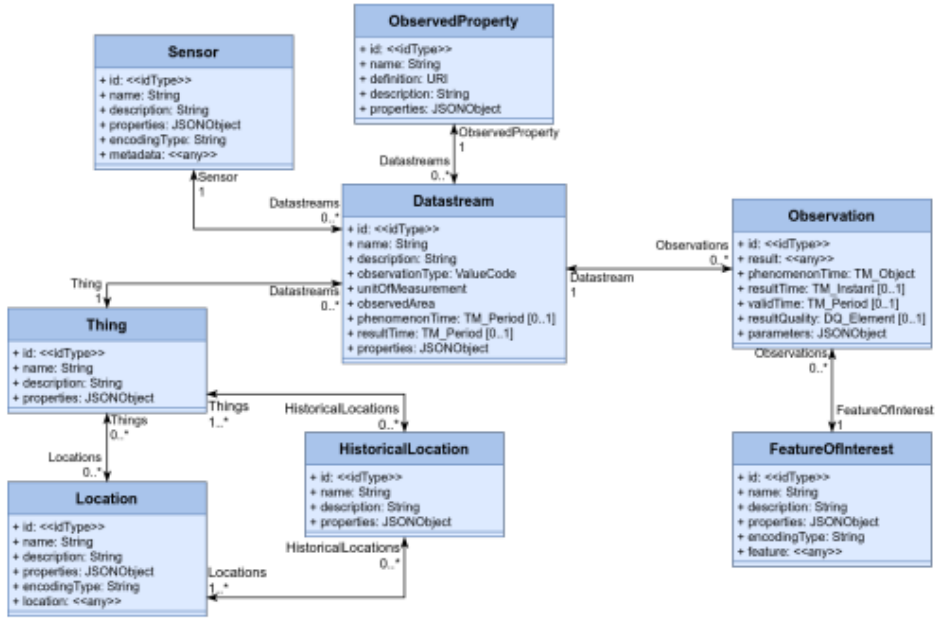
Based on Observations & Measurements

- Easy to use & understandable
  - Discoverable with only a web browser

OGC SensorThings API Part 1 - context

How does it work?

- Part 1: Data model
  - Which entities exist
  - How are they linked



- Part 2: URL patterns for queries
  - How do I get & search data
  - How do I add data
  - How do I modify data
  - How do I delete data

REST
&
MQTT

OGC SensorThings API Part 1 - context

Implementations

Scroll to Organization: -

52 North GmbH Top

52N Helgoland API 3.3.3	Jirka, Simon	Registered: 2016-08-10

Fraunhofer-Gesellschaft Top

FROST-Server 1.14.x OGC SensorThings API Part 1: Sensing 1.0 (Certified: 2021-11-01)	Hylke van der Schaaf	Registered: 2016-08-10

Geodan Holding BV Top

GOST 0.3	Bert Temme	Registered: 2016-11-28
GOST 0.6	Bert Temme	Registered: 2018-01-10
GOST 0.6.1	Bert Temme	Registered: 2020-01-28

i-bitz company limited Top

Vallaris Maps 2020	Pongsakorn Udombua	Registered: 2021-09-06

Institute of Communication and Computer Systems (ICCS) Top

SensorThings 1.0 OGC SensorThings API Part 1: Sensing 1.0 (Certified: 2019-06-06)	Fay Misichroni	Registered: 2019-05-30

NCU Top

SensorThings-MongoDB 1	Ting, Yu Yeh	Registered: 2022-06-27

PilotGaea Technologies Co., Ltd. Top

PilotGaea GIS 11.0	LI, Yashu	Registered: 2020-01-07
PilotGaea GIS 12.0	LI, Yashu	Registered: 2021-01-17
PilotGaea GIS 13.0 OGC SensorThings API Part 1: Sensing 1.0 (Certified: 2022-02-11)	LI, Yashu	Registered: 2022-02-21

SensorUp Inc. Top

SensorUp SensorThings API v1.0	Steve Liang	Registered: 2016-02-29
SensorUp SensorThings API v1.0	Steve Liang	Registered: 2016-02-29

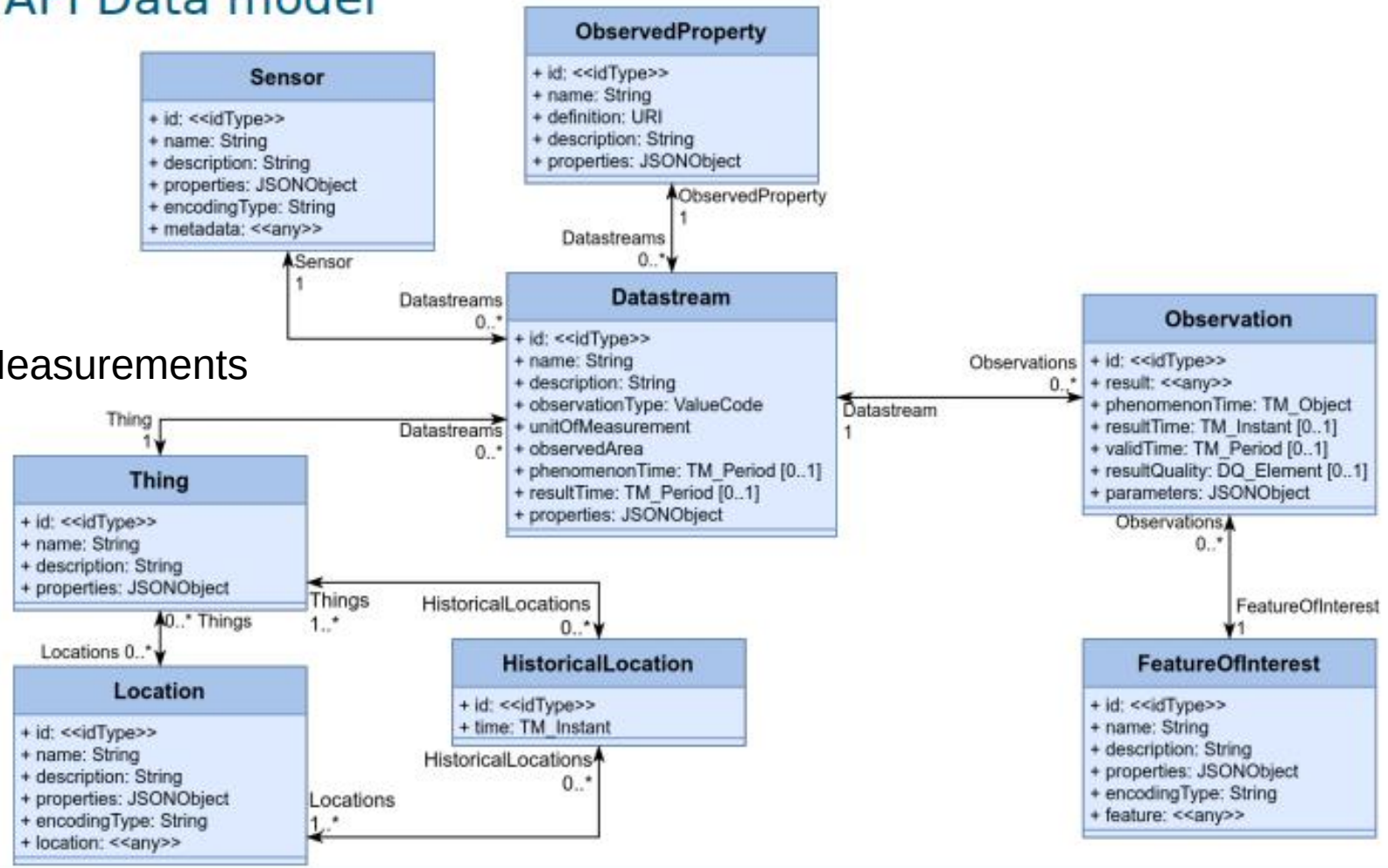
https://www.ogc.org/resource/products/byspec/?specid=772

OGC SensorThings API Part 1 – data model

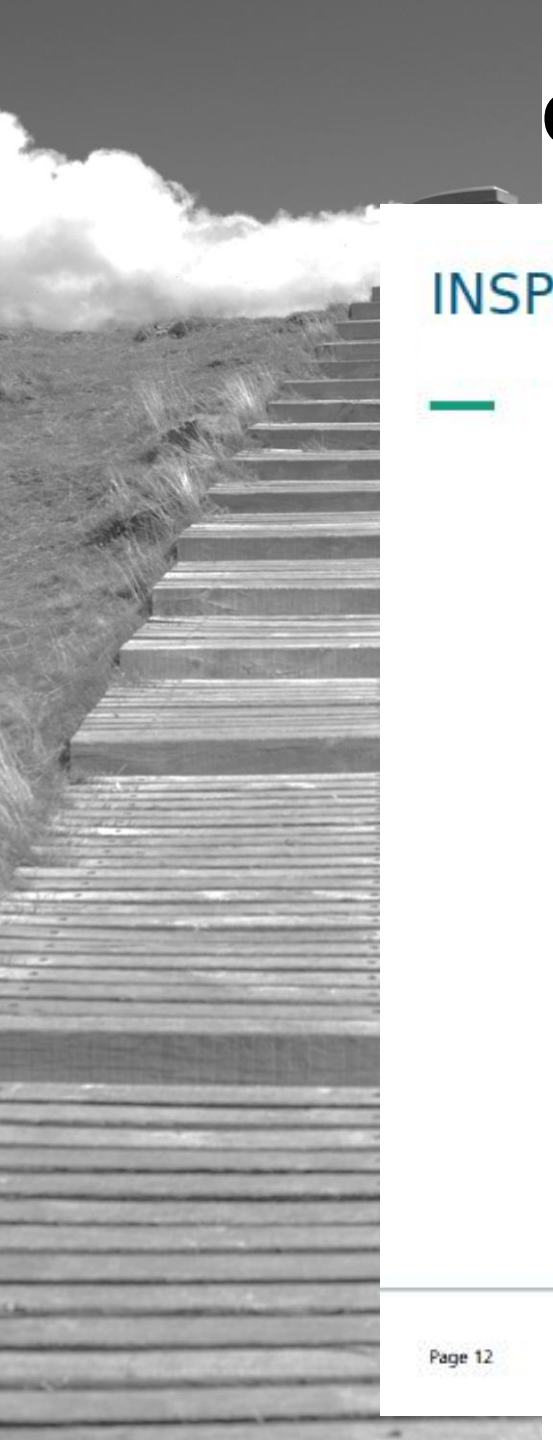
SensorThing API Data model

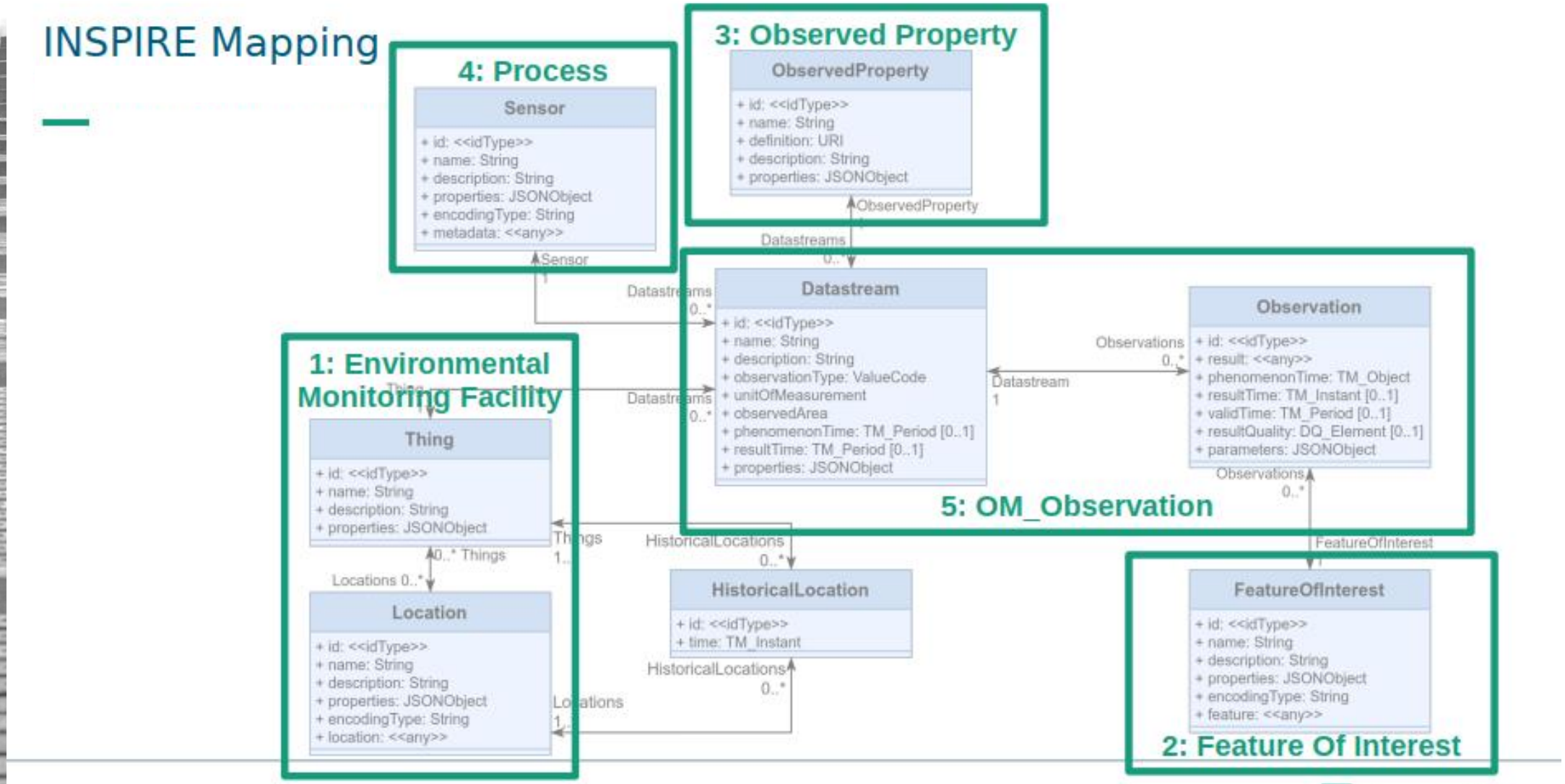
The Core

Based on
Observations & Measurements



OGC SensorThings API Part 1 – data model

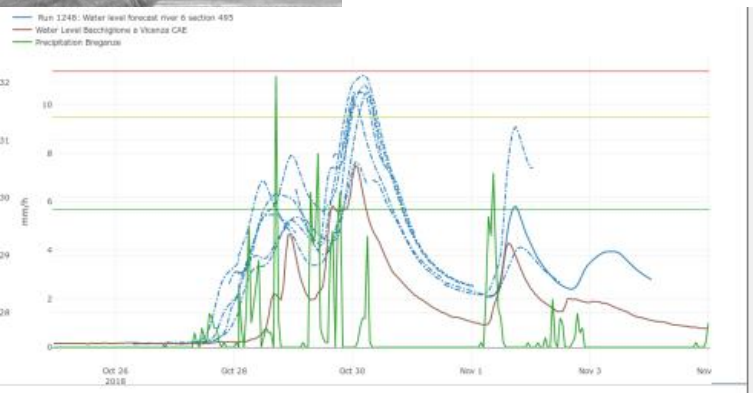


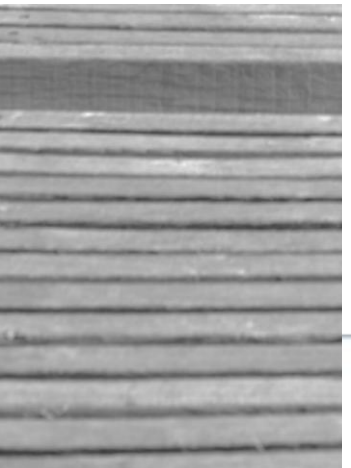


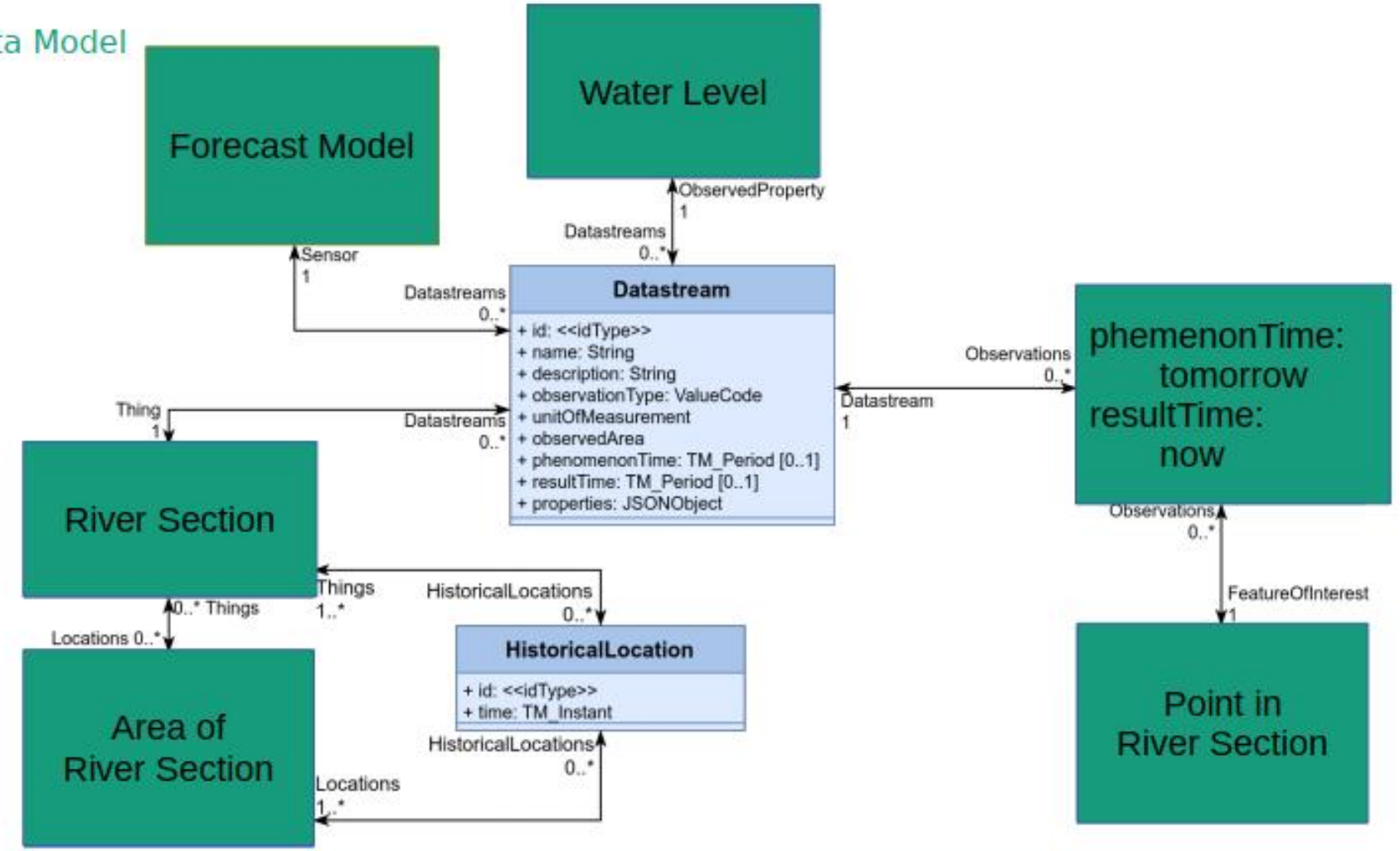
OGC SensorThings API Part 1 – data model

BeAWARE

Flood Scenario Data Model

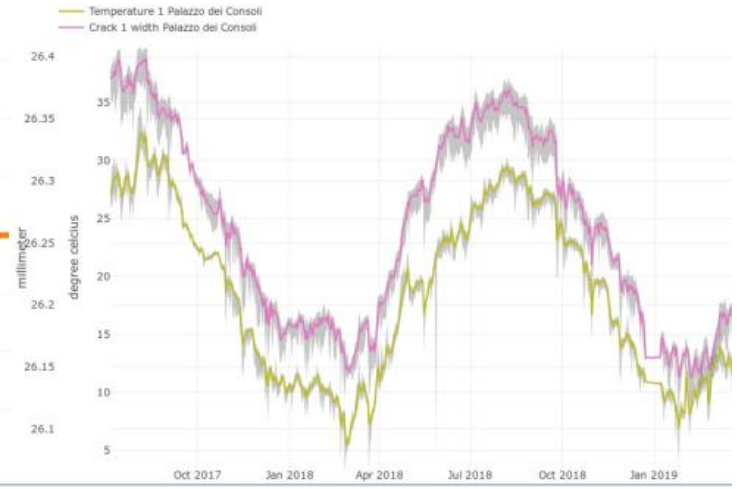


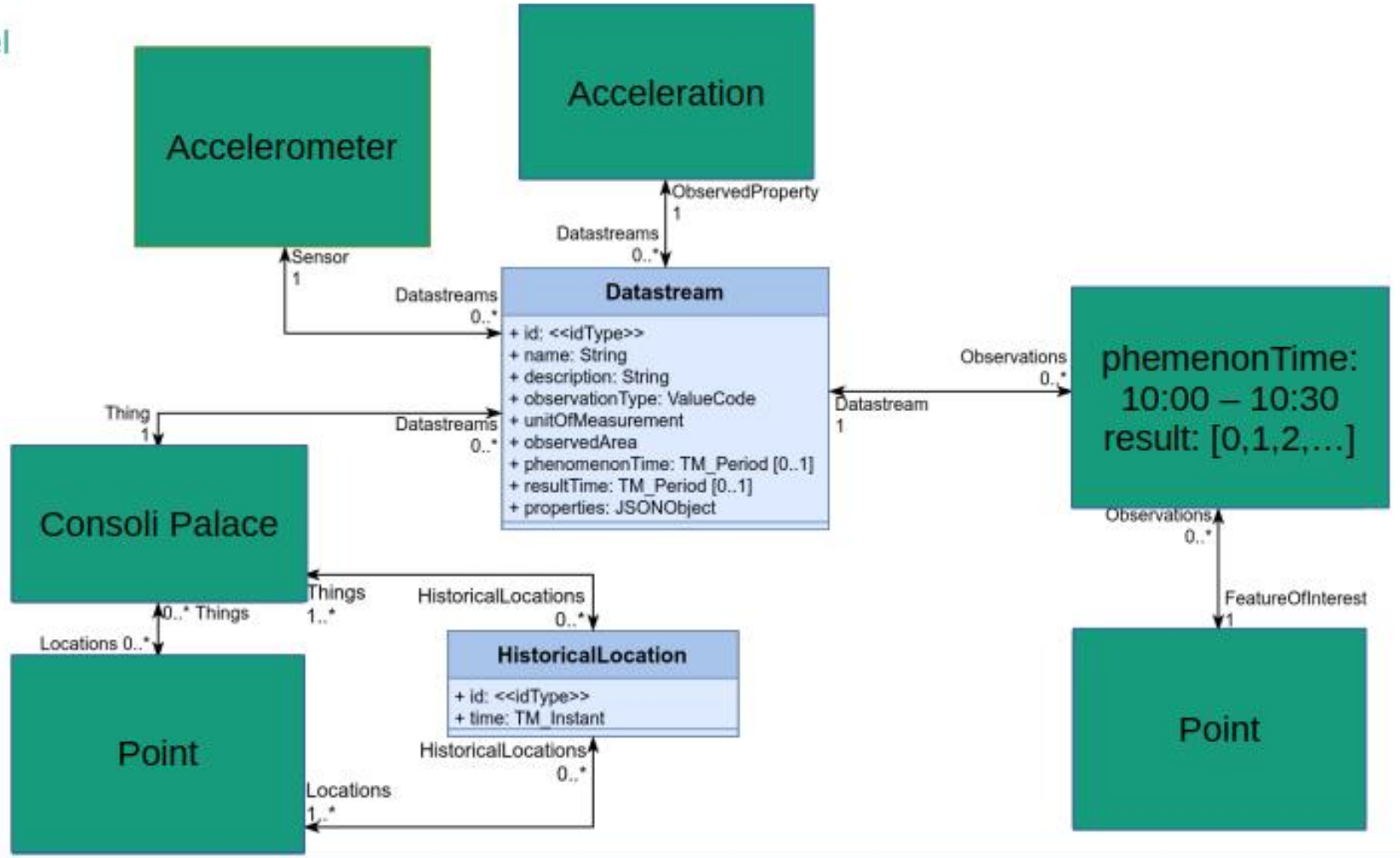




OGC SensorThings API Part 1 – data model

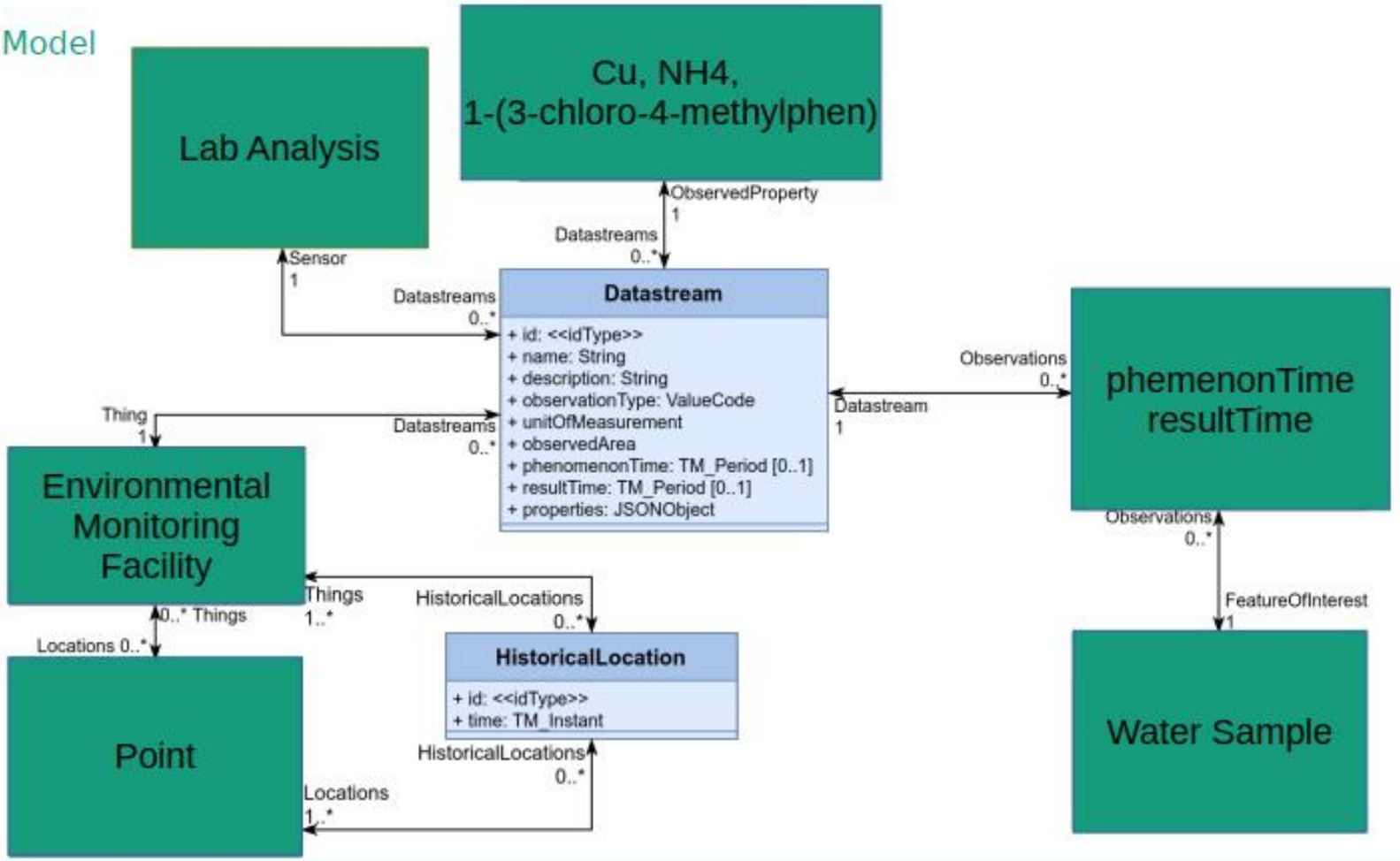
HERACLES
Building Data Model

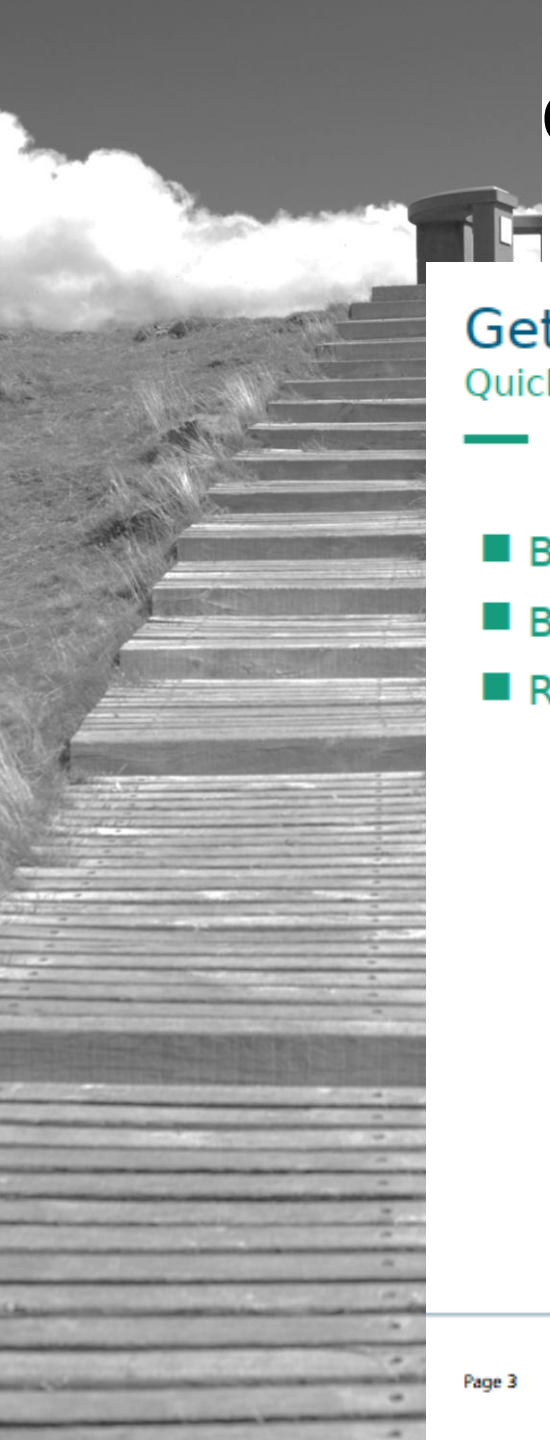




OGC SensorThings API Part 1 – data model

BRGM
Water Quality Data Model





OGC SensorThings API Part 1 – REST API

Getting to your data

Quick Overview

- Based on OASIS OData
- Base URL: http://server.de/FROST-Server/v1.1
- Read: GET
  - v1.1 → Get index
  - v1.1/Collection → Get all in a set
  - v1.1/Collection(id) → Get one from a set
- Create: POST
  - v1.1/Collection → Create a new entity
- Update: PATCH
  - v1.1/Collection(id) → Update an entity
- Update: PUT
  - v1.1/Collection(id) → Replace an entity
- Delete: DELETE
  - v1.1/Collection(id) → Remove an entity

OGC SensorThings API Part 1 – REST API

Query URL patterns: Index

Get Service Index

- GET http://ogctest.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1
  - Response:


```
{
  "value" : [
    {
      "name" : "Datastreams",
      "url" : "http://server.de/SensorThingsService/v1.0/Datastreams"
    },
    {
      "name" : "FeaturesOfInterest",
      "url" : "http://server.de/SensorThingsService/v1.0/FeaturesOfInterest"
    },
    {
    …
    …
    },
    {
      "name" : "Things",
      "url" : "http://server.de/SensorThingsService/v1.0/Things"
    }
  ],
  "serverSettings": {}
}
```


OGC SensorThings API Part 1 – REST API

Query URL patterns: Get Collection

Get All Things

- GET http://ogctest.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Things
  - Response:

```
{
  "value" : [
    {
      "name" : "My camping lantern",
      "description" : "camping lantern",
      "properties" : {
        "property1" : "it's waterproof",
        "property2" : "it glows in the dark"
      },
      "Locations@iot.navigationLink" : "Things(1)/Locations",
      "HistoricalLocations@iot.navigationLink":
"Things(1)/HistoricalLocations",
      "Datastreams@iot.navigationLink" : "Things(1)/Datastreams",
      "@iot.id" : 1,
      "@iot.selfLink" : "/SensorThingsService/v1.0/Things(1)"
    },
    {
      a second thing…
    }, { … }, { … }, { … }
  ]
}
```

OGC SensorThings API Part 1 – REST API

Query URL patterns: Get one Entity

Get Specific Thing

- GET http://ogctest.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Things(1)
  - Response:

```
{
  "name" : "My camping lantern",
  "description" : "camping lantern",
  "properties" : {
    "property1" : "it's waterproof",
    "property2" : "it glows in the dark"
  },
  "Locations@iot.navigationLink" : "Things(1)/Locations",
  "HistoricalLocations@iot.navigationLink" : "Things(1)/HistoricalLocations",
  "Datastreams@iot.navigationLink" : "Things(1)/Datastreams",
  "@iot.id" : 1,
  "@iot.selfLink" : "/SensorThingsService/v1.0/Things(1)"
}
```

OGC SensorThings API Part 1 – REST API

Query URL patterns: Get one Entity

Get Specific Thing

- GET http://ogctest.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Things(1)
  - Response:

```
{
  "name" : "My camping lantern",
  "description" : "camping lantern",
  "properties" : {
    "property1" : "it's waterproof",
    "property2" : "it glows in the dark"
  },
  "Locations@iot.navigationLink" : "Things(1)/Locations",
  "HistoricalLocations@iot.navigationLink" : "Things(1)/HistoricalLocations",
  "Datastreams@iot.navigationLink" : "Things(1)/Datastreams",
  "@iot.id" : 1,
  "@iot.selfLink" : "/SensorThingsService/v1.0/Things(1)"
}
```

OGC SensorThings API Part 1 – REST API

Lot’s of other possibilities

- Get Related Entities
- Pagination
- Define the content you need in the response
  - ‘Select’: to reduce it to the content you need
  - ‘Expand’: to add information from relater entities
- Sorting
- Filtering
- And (next slide)

OGC SensorThings API Part 1 – REST API

Query URL patterns: Functions 1

Lots of Choice

- Comparison:
  - gt: > ge: >=
  - Eq: = le: <=
  - lt: < ne: !=
- Logical:
  - and / or / not
- Mathematical:
  - add / sub / mul / div / mod
  - round(n1)
  - floor(n1) / ceiling(n1)

- String Functions:
  - substringof(p0, p1)
  - endswith(p0, p1)
  - startswith(p0, p1)
  - substring(p0, p1)
  - indexof(p0, p1)
  - length(p0)
  - tolower(p0)
  - toupper(p0)
  - trim(p0)
  - concat(p0, p1)

OGC SensorThings API Part 1 – REST API

Query URL patterns: Functions 2

Even more choice!

- Geospatial:
  - geo.intersects(g1, g2)
  - geo.length(l1)
  - geo.distance(g1, g2)
  - st_equals(g1, g2)
  - st_disjoint(g1, g2)
  - st_touches(g1, g2)
  - st_within(g1, g2)
  - st_overlaps(g1, g2)
  - st_crosses(g1, g2)
  - st_intersects(g1, g2)
  - st_contains(g1, g2)
  - st_relate(g1, g2)
- Date and Time:
  - now()
  - mindatetime()
  - maxdatetime()
  - date(t1)
  - time(t1)
  - year(t1)
  - month(t1)
  - day(t1)
  - hour(t1)
  - minute(t1)
  - second(t1)
  - fractionalseconds(t1)
  - totaloffsetminutes(t1)

OGC SensorThings API Part 1 – REST API

Query URL patterns: Filtering examples

Find me a Thing

- All observations with an even result
  - Observations?$filter=result mod 2 eq 0
- Observations of the last hour
  - Observations?$filter=phenomenonTime gt now() sub duration’PT1H’
  - https://en.wikipedia.org/wiki/ISO_8601#Durations
- Datastreams that measure temperature (ObservedProperty id 1)
  - Datastreams?$filter=ObservedProperty/@iot.id eq 1
- Filtering on JSON properties
  - Things?$filter=properties/style eq 'Cozy'
  - Observations?$filter=result gt Datastream/properties/max

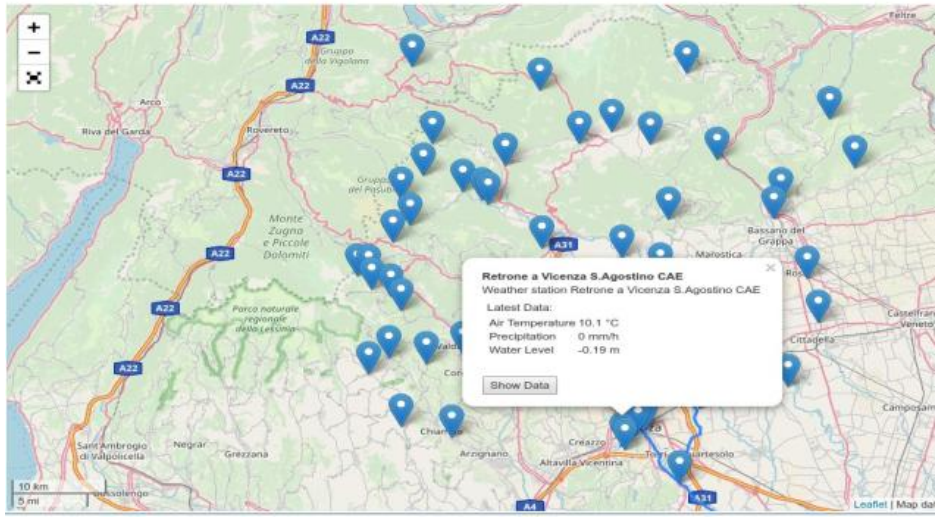
OGC SensorThings API Part 1 – REST API

Query URL patterns: $expand example

Get everything for my map in 1 request

```
…/v1.1/Things?
    $select=id,name,description,properties
    &$top=1000
    &$filter=properties/type eq 'station'
    &$expand=
        Locations,
        Datastreams(
            $select=id,name,unitOfMeasurement
            ;$expand=
                ObservedProperty($select=name),
                Observations(
                    $select=result,phenomenonTime
                    ;$orderby=phenomenonTime desc
                    ;$top=1)
        )
```

Overview Map (Italy)



OGC SensorThings API Part 1 – REST API

Creating new Entities

Create a new Thing

- POST … /v1.1/Things

```
Content-Type: application/json;charset=UTF-8

{
  "name" : "Office",
  "description" : "My Work Room",
  "properties" : {
    "style" : "Business",
    "balcony" : false
  },
  "Locations" : [
    {
      "@iot.id" : 1
    }
  ]
}
```

  - Response:

```
Location: http://localhost:8080/FROST-Server/v1.1/Things(2)
```

OGC SensorThings API Part 1 – REST API

Creating new Observations

Create a new Observation

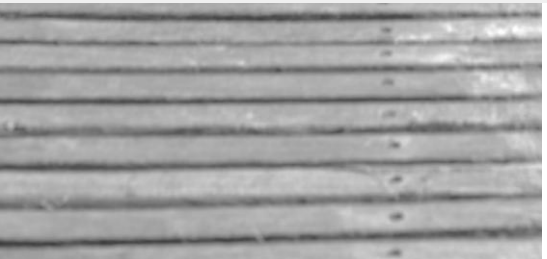
- POST … /v1.1/Observations

```
{
  "result" : 123,
  "Datastream" : {
    "@iot.id" : 1
  }
}
```

- POST … /v1.1/Datastreams(1)/Observations

```
{
  "result" : 123
}
```

- phenomenonTime and FeatureOfInterest are generated automatically if not provided.



+ changing, deleting entities
+ extensions

Creating new Observations – HTTP vs MQTT

Options, options …

	+	–
MQTT	• Efficient for subsequent Observations	• Connection management • Can only create Observations • Persistent connection makes load balancing tricky
HTTP	• Simple requests • Can create all entities • Works through firewalls & proxies • Simple load balancing	• New connection per request

OGC SensorThings API Part 1 – marine example

French Marine Agency (IFREMER)

La modélisation des objets métiers

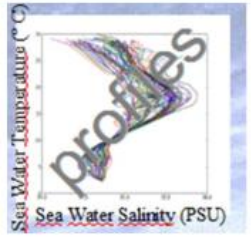
Les profils (ex : flotteurs Argo) -> un cas particulier, quand le X n'est pas le temps

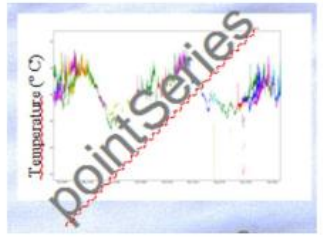
Un seul temps pour n observations selon Z (la profondeur)

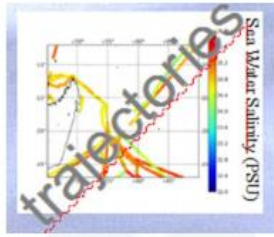
7

Le contexte métier

- Profiles: CTD, profiling floats (ARGO)
- PointSeries: moorings, including moorings with different vertical levels
- Trajectories: thermo-salinomers, gliders



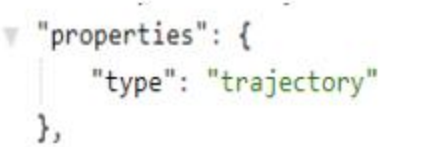


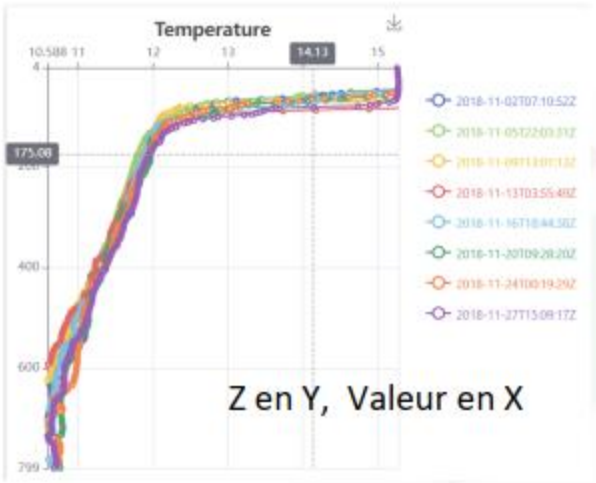


3

OGC SensorThings API Part 1 – marine example

French Marine Agency (IFREMER)

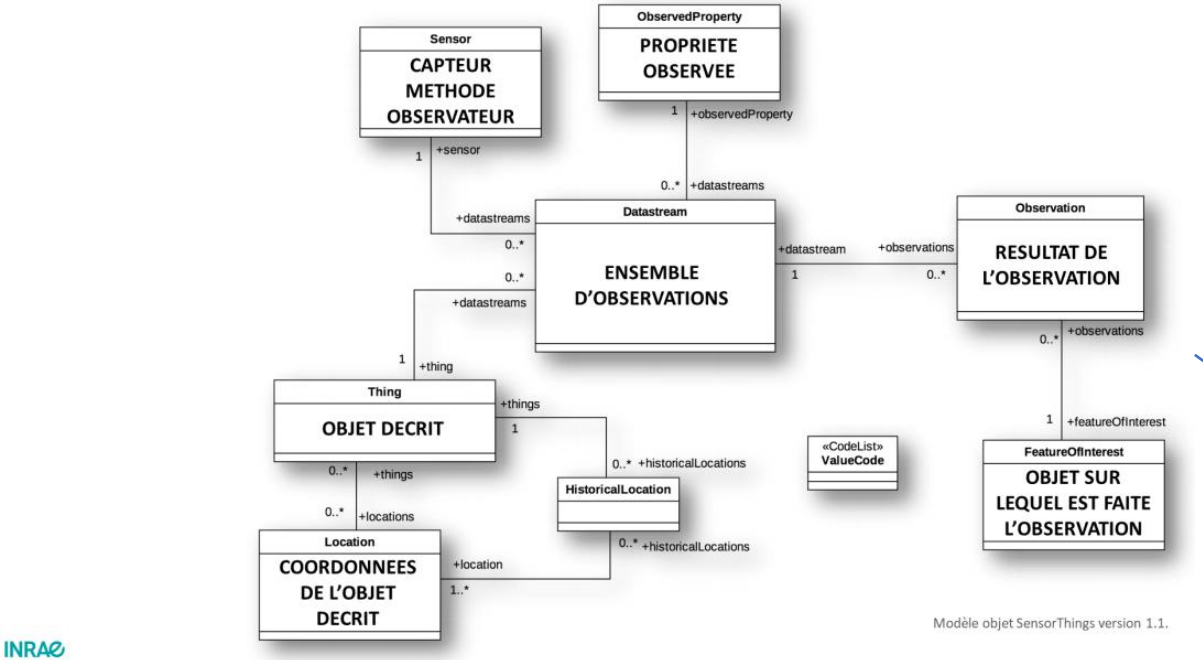




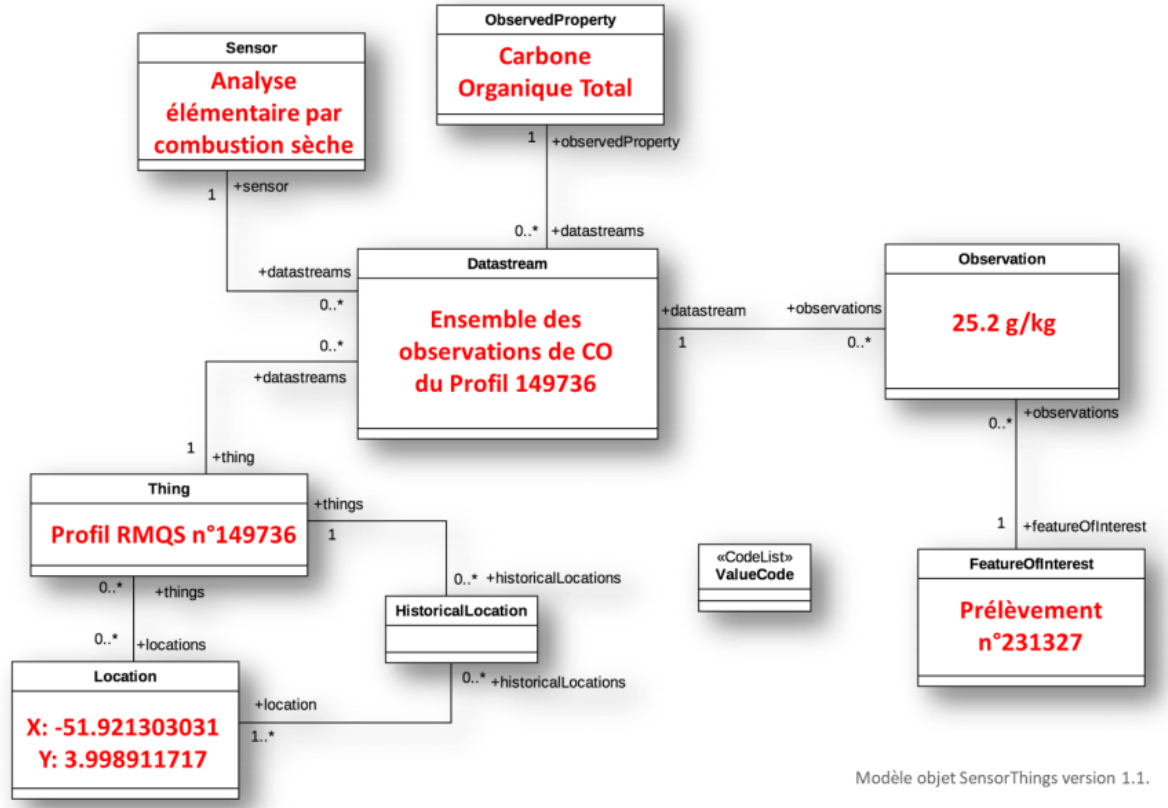
https://sextant.ifremer.fr/examind/WS/sts/coriolis/v1.1

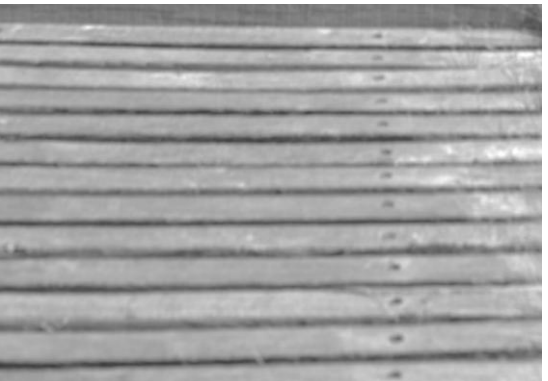
OGC SensorThings API Part 1 – Agronomy example

French Agronomic Institute (INRAE)



Organic Carbon in a soil sample

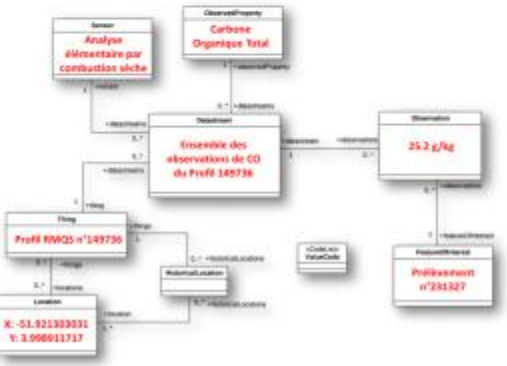


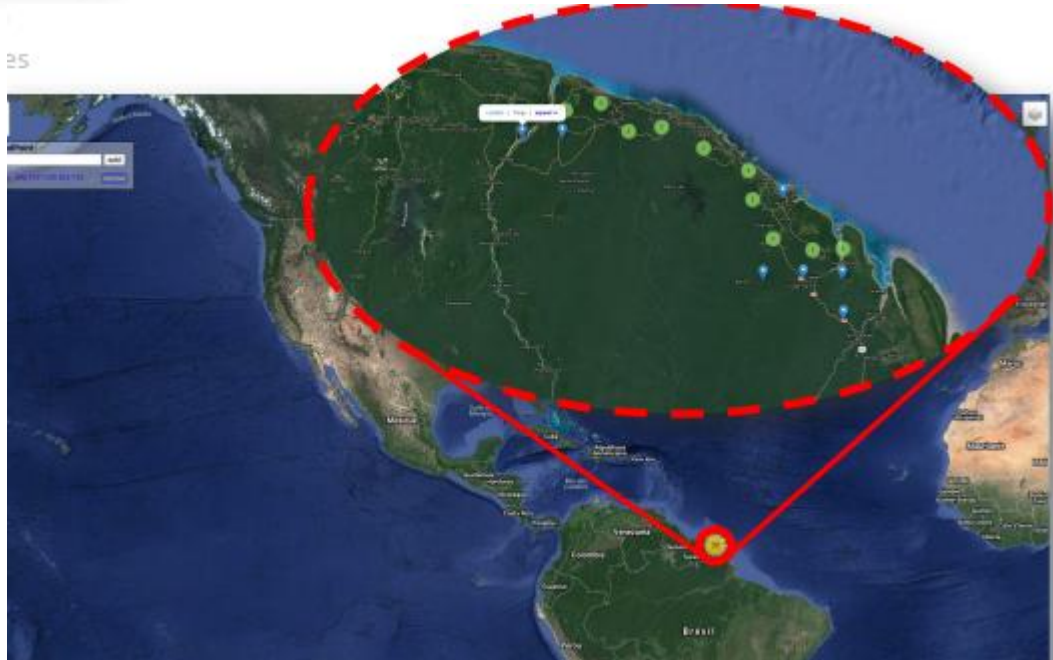


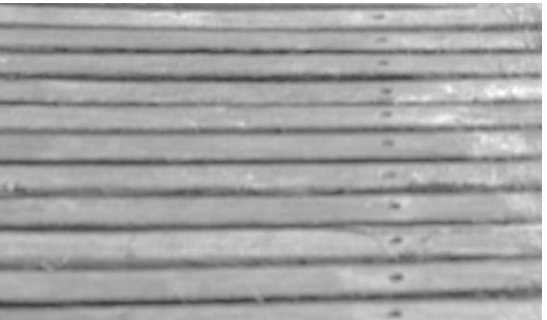
OGC SensorThings API Part 1 – Agronomy example

French Agronomic Institute (INRAE)

- Historical/Legacy national databank mapped to SensorThings API for data sharing

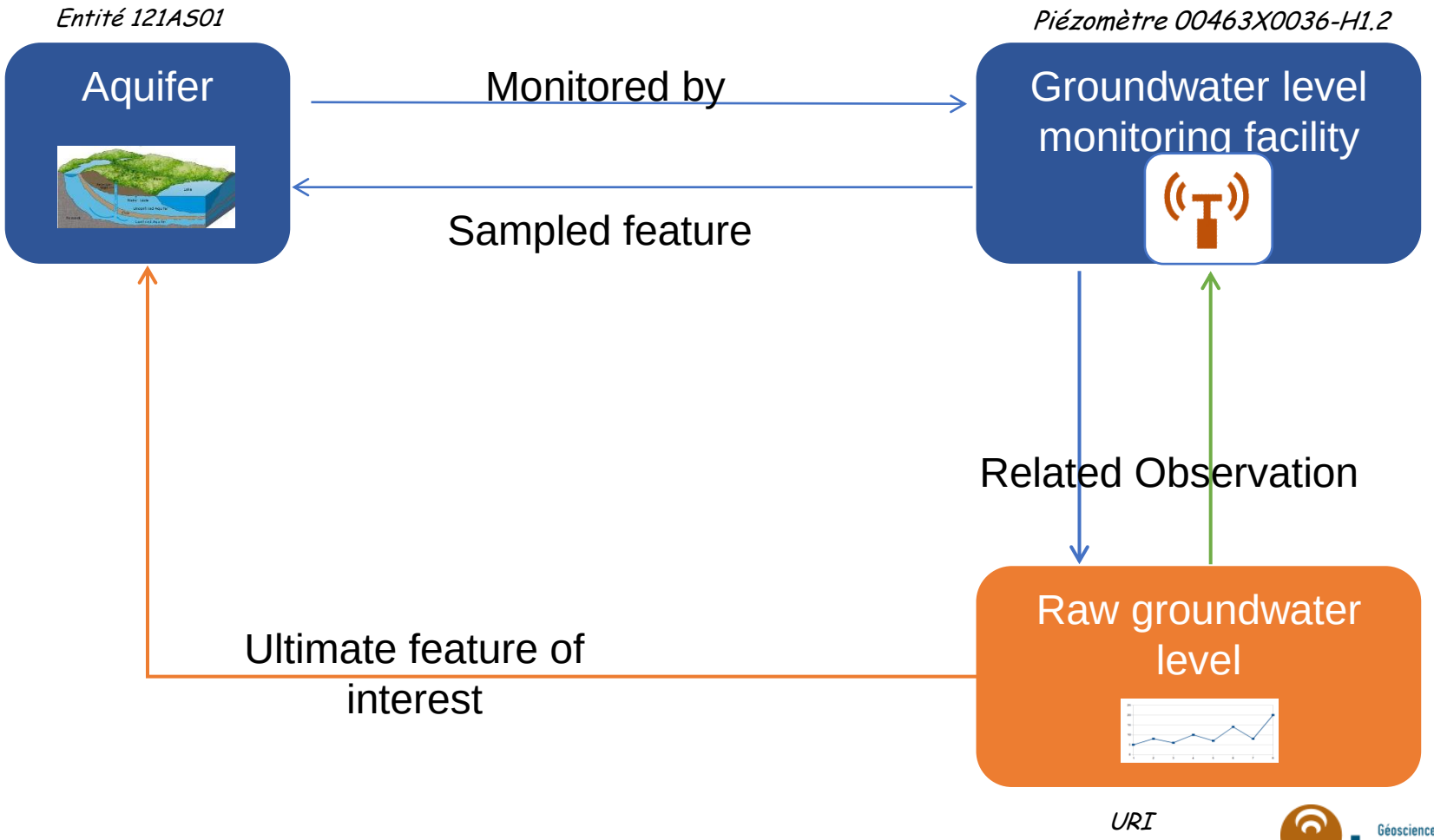






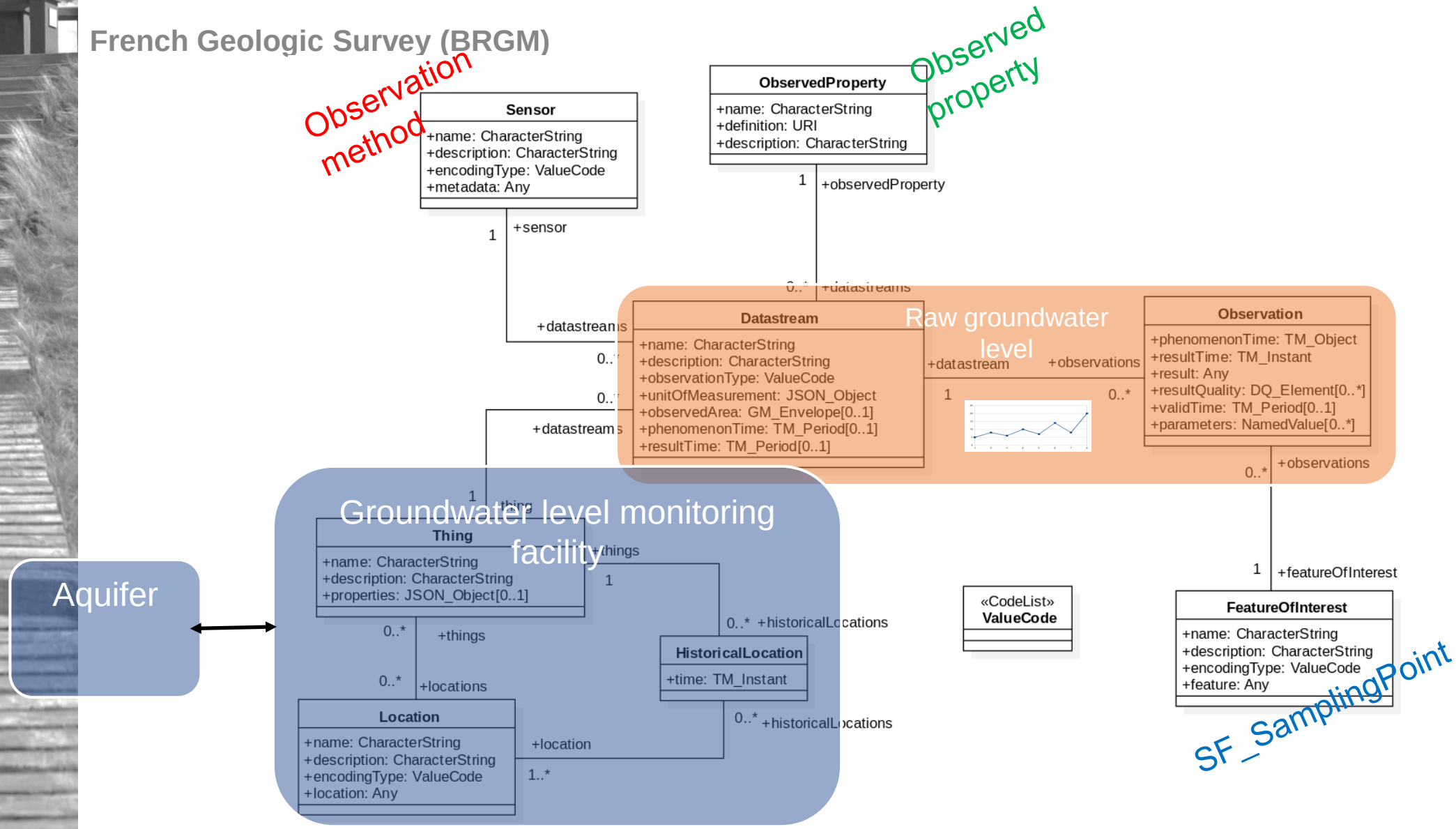
OGC SensorThings API Part 1 – GroundWater example

French Geologic Survey (BRGM)



OGC SensorThings API Part 1 – GroundWater example

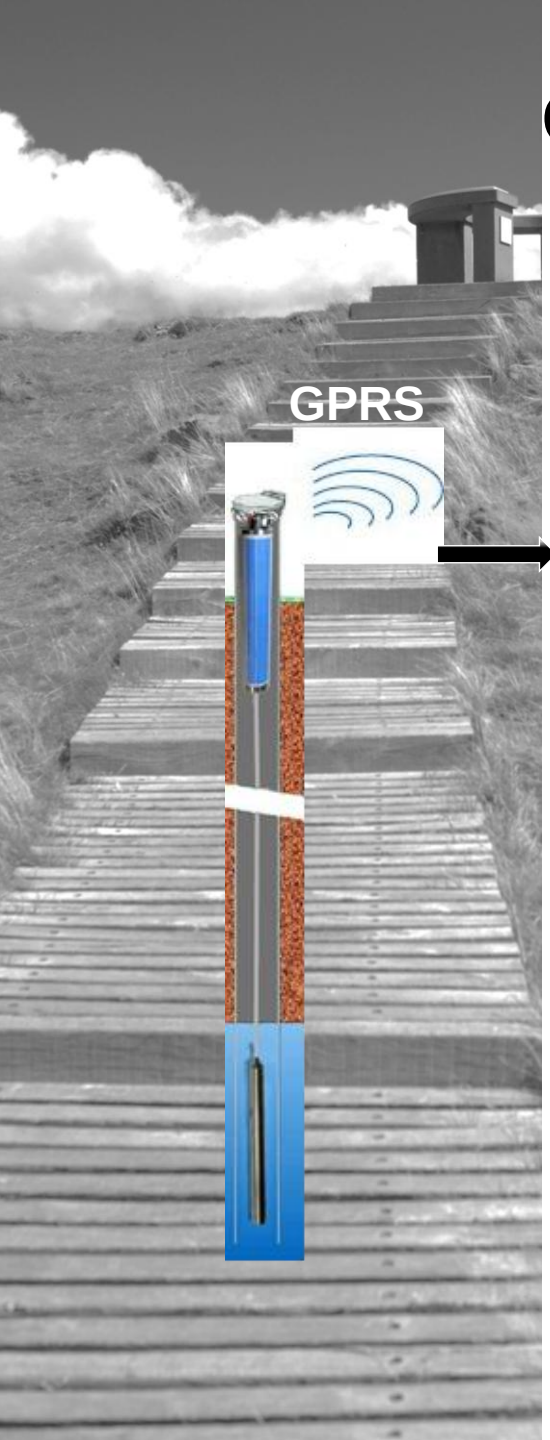
French Geologic Survey (BRGM)

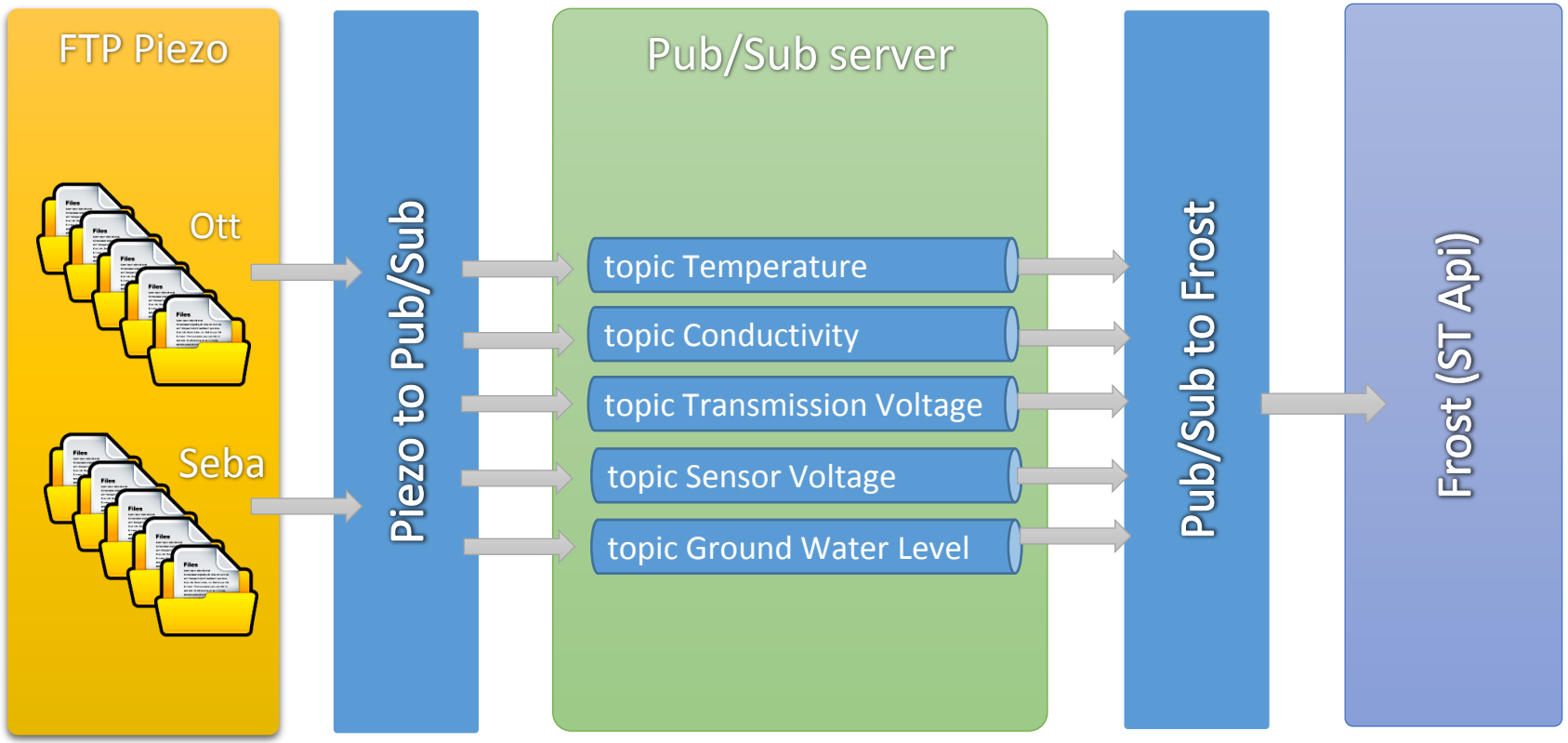


OGC SensorThings API Part 1 – GroundWater example

French Geologic Survey (BRGM)

- From field sensors to SensorThings API server



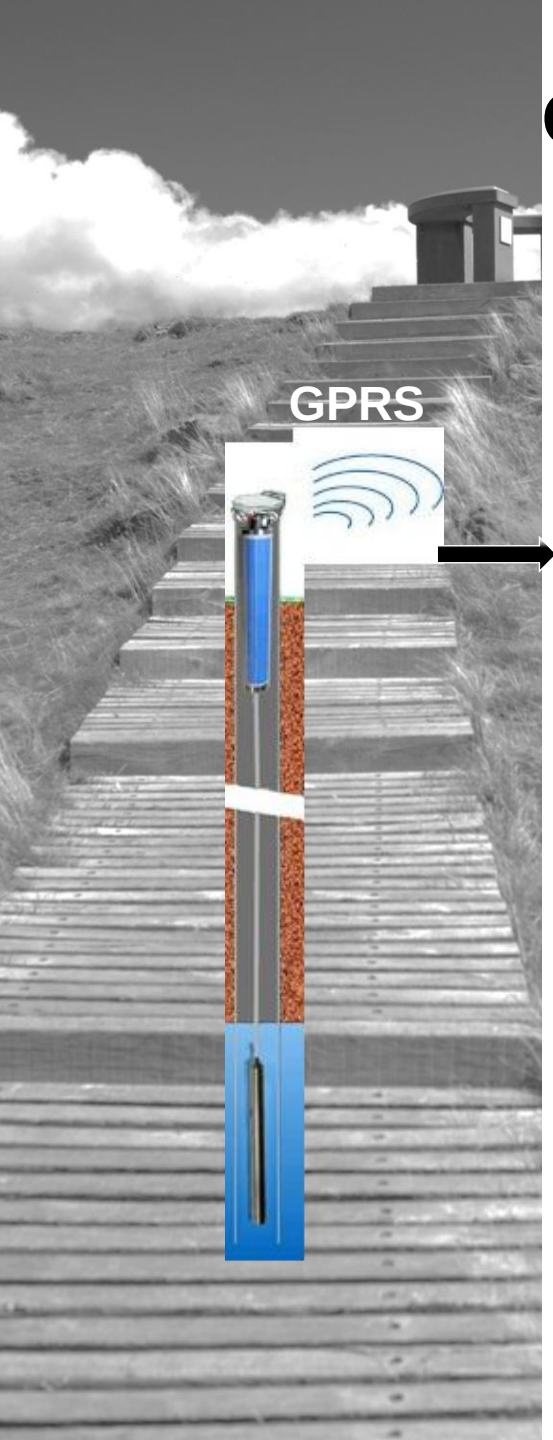


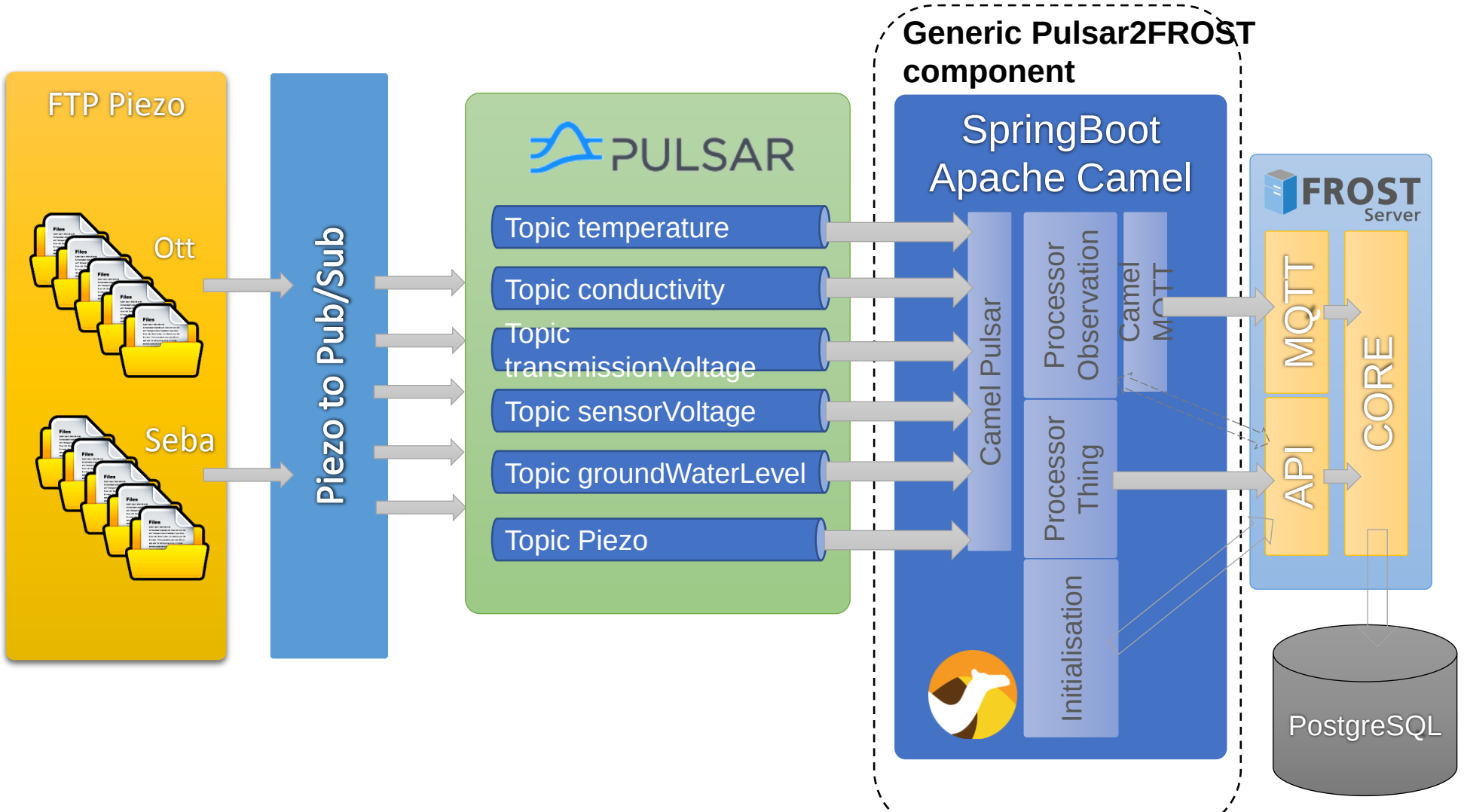
Géosciences pour une Terre durable
brgm

OGC SensorThings API Part 1 – GroundWater example

French Geologic Survey (BRGM)

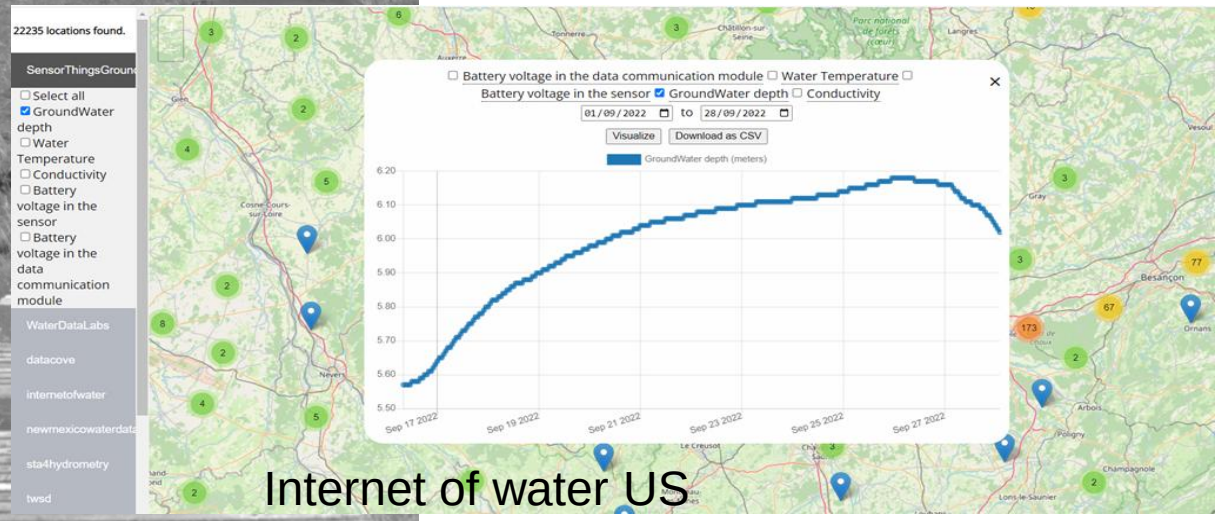
- From field sensors to SensorThings API server using the full RESTful and MQTT capacities of the standard



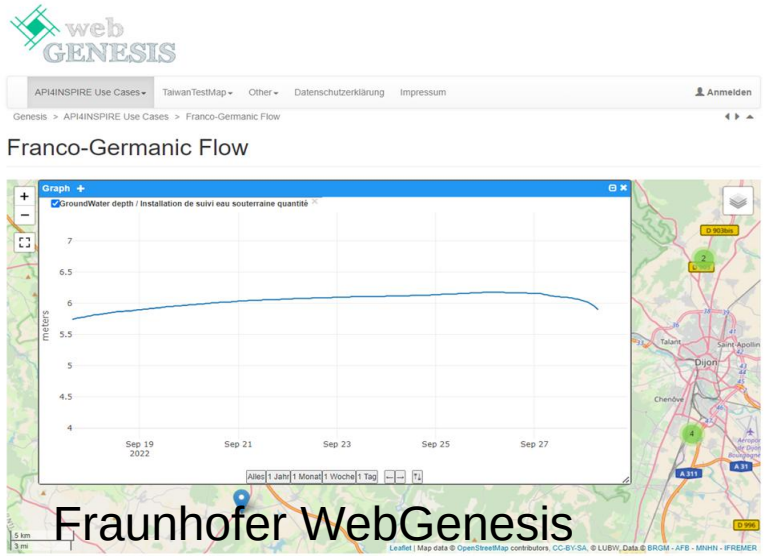


OGC SensorThings API Part 1 – client side

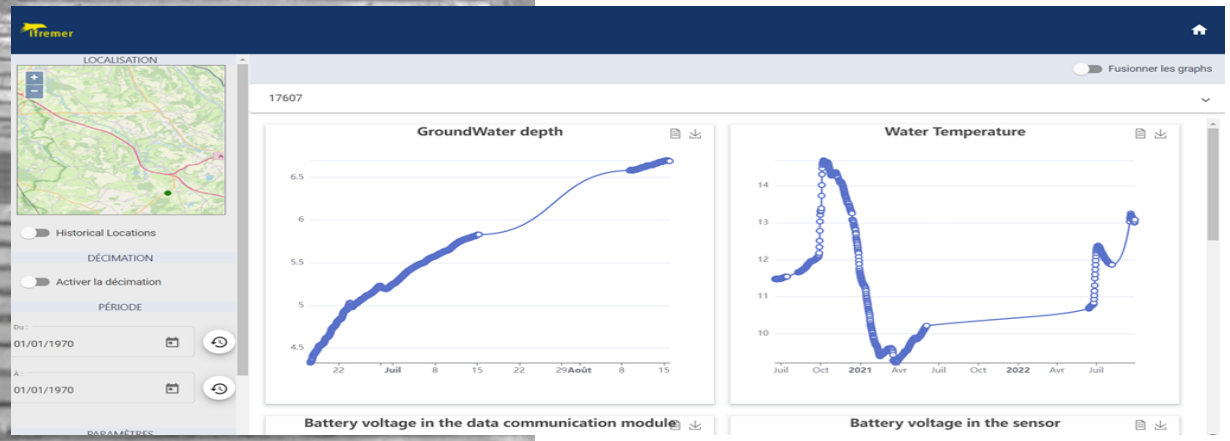
One ground water endpoint (BRGM) various clients



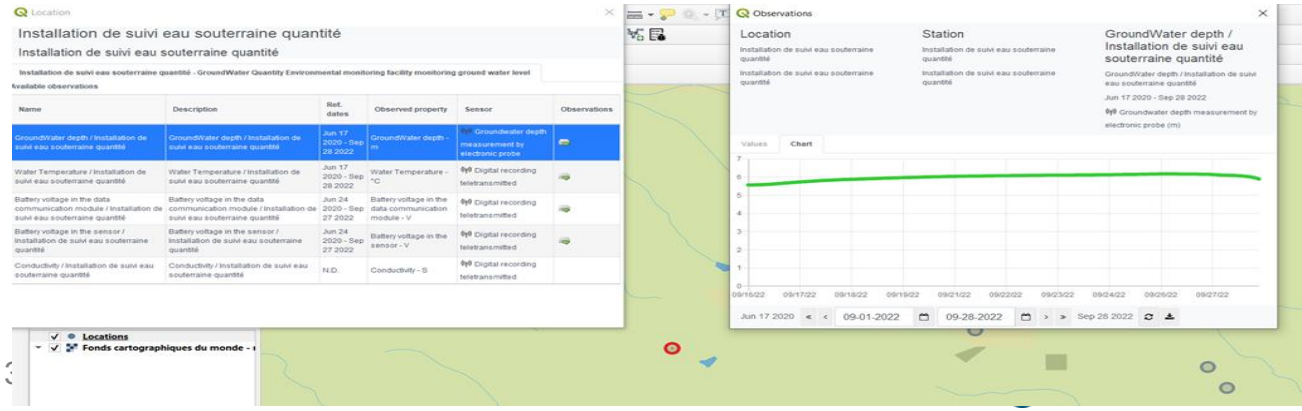
Internet of water US



Fraunhofer WebGenesis



Geomatys Examind Community



QGIS new SensorThings API plugin (soon ported to Core)

Extra material

- Observations, Measurement And Samples
  - https://github.com/opengeospatial/om-swg
  - New version available soon here : https://www.ogc.org/standards/om

- SensorThings API Part 1
  - Standard : https://www.ogc.org/standards/sensorthings
  - https://github.com/opengeospatial/sensorthings
  - Many screenshots come from this workshop (EN/French) :
    - https://github.com/INSIDE-information-systems/SensorThingsAPI/tree/master/presentations/20220929_INSIDE_SIST_SensorThingsAPI_Webinaire

Thank you

s.grellet@brgm.fr

Thanks to

Kathi Schleidt

Kathi@DataCove.eu

Hylke van den Schaaf

hylke.vanderschaaf@iosb.fraunhofer.de