SensorThings

Luca Giovannini

Many thanks to Francesca Noardo from OGC…

Open Geospatial Consortium (OGC)

deda.next
x EDIAQI

OGC is a no-profit international membership organisation committed to improving access to geospatial, or location information through the development of Standards.

Findable **A**ccessible **I**nteroperable **R**eusable

Community

500+ International Members
110+ Member Meetings
60+ Alliance and Liaison partners
50+ Standards Working Groups
45+ Domain Working Groups
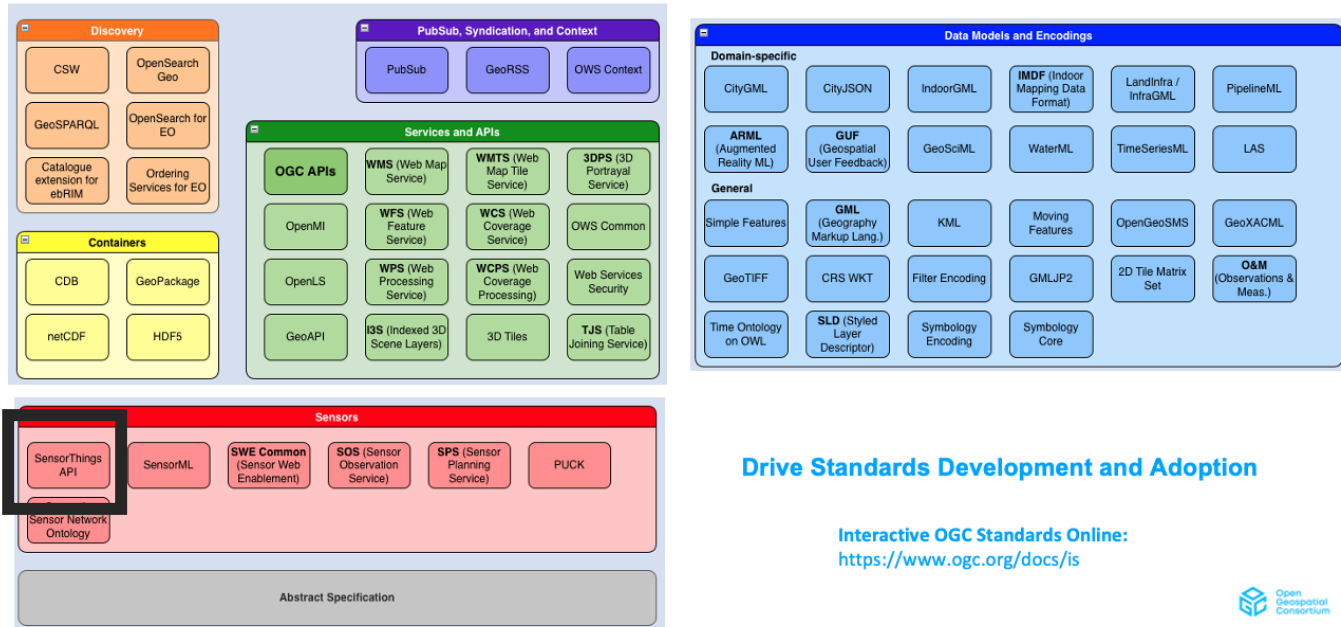25+ Years of Not for Profit Work
10+ Regional and Country Forums

Standards

65+ Adopted Standards
300+ products with 1000+ certified implementations
1,700,000+ Operational Data Sets
Using OGC Standards

Innovation

120+ Innovation Initiatives
380+ Technical reports
Quarterly Tech Trends monitoring

Standards Architecture Diagram

Discovery

CSW | OpenSearch Geo
GeoSPARQL | OpenSearch for EO
Catalogue extension for ebRIM | Ordering Services for EO

Containers

CDB | GeoPackage
netCDF | HDF5

PubSub, Syndication, and Context

PubSub | GeoRSS | OWS Context

Services and APIs

OGC APIs | WMS (Web Map Service) | WMTS (Web Map Tile Service) | 3DPS (3D Portrayal Service)
OpenMI | WFS (Web Feature Service) | WCS (Web Coverage Service) | OWS Common
OpenLS | WPS (Web Processing Service) | WCPS (Web Coverage Processing) | Web Services Security
GeoAPI | I3S (Indexed 3D Scene Layers) | 3D Tiles | TJS (Table Joining Service)

Data Models and Encodings

Domain-specific

CityGML | CityJSON | IndoorGML | IMDF (Indoor Mapping Data Format) | LandInfra / InfraGML | PipelineML
ARML (Augmented Reality ML) | GUF (Geospatial User Feedback) | GeoSciML | WaterML | TimeSeriesML | LAS

General

Simple Features | GML (Geography Markup Lang.) | KML | Moving Features | OpenGeoSMS | GeoXACML
GeoTIFF | CRS WKT | Filter Encoding | GMLJP2 | 2D Tile Matrix Set | O&M (Observations & Meas.)
Time Ontology on OWL | SLD (Styled Layer Descriptor) | Symbology Encoding | Symbology Core

Sensors

SensorThings API | SensorML | SWE Common (Sensor Web Enablement) | SOS (Sensor Observation Service) | SPS (Sensor Planning Service) | PUCK
Sensor Network Ontology

Abstract Specification

Drive Standards Development and Adoption

Interactive OGC Standards Online:
https://www.ogc.org/docs/is

Open Geospatial Consortium

Open Geospatial Consortium (OGC)

OGC is a no-profit international membership organisation committed to improving access to geospatial, or location information through the development of Standards.

Community

500+ International Members

110+ Member Meetings

60+ Alliance and Liaison partners

50+ Standards Working Groups

45+ Domain Working Groups

25+ Years of Not for Profit Work

10+ Regional and Country Forums

Standards

65+ Adopted Standards

300+ products with 1000+ certified implementations

1,700,000+ Operational Data Sets

Using OGC Standards

Innovation

120+ Innovation Initiatives

380+ Technical reports

Quarterly Tech Trends monitoring

Findable   Accessible   Interoperable   Reusable

OGC SensorThings API

https://www.ogc.org/standard/sensorthings/

Documentation:
https://docs.ogc.org/is/15-078r6/15-078r6.html

Internet of Things

Internet of Things (**IoT**) describes physical objects (or groups of such objects) with sensors, processing ability, software and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks.

https://en.wikipedia.org/wiki/Internet_of_things

IoT device = sensors and actuators connected to a network and able to transmit data.

Application Programming Interface (API) is a way for two or more computer programs to communicate with each other. It is a type of software interface, offering a service to other pieces of software

https://en.wikipedia.org/wiki/API

OGC SensorThings API

Advantages of OGC SensorThings API

Most IoT devices today have **proprietary software interfaces** defined by manufacturers and used selectively.

➡️ **New APIs** are therefore requested and developed as needed, often in an environment with **limited resources** and associated **risks**.

➡️ **Relevant investments** for each new sensor or project involving multiple systems.

As a standardized data model and interface for IoT sensors, the OGC SensorThings API offers the following benefits:

1. **enables** the proliferation of **new high-value services** with lower development overhead and broader reach

2. **reduces risk, time and cost** in a complete IoT product cycle

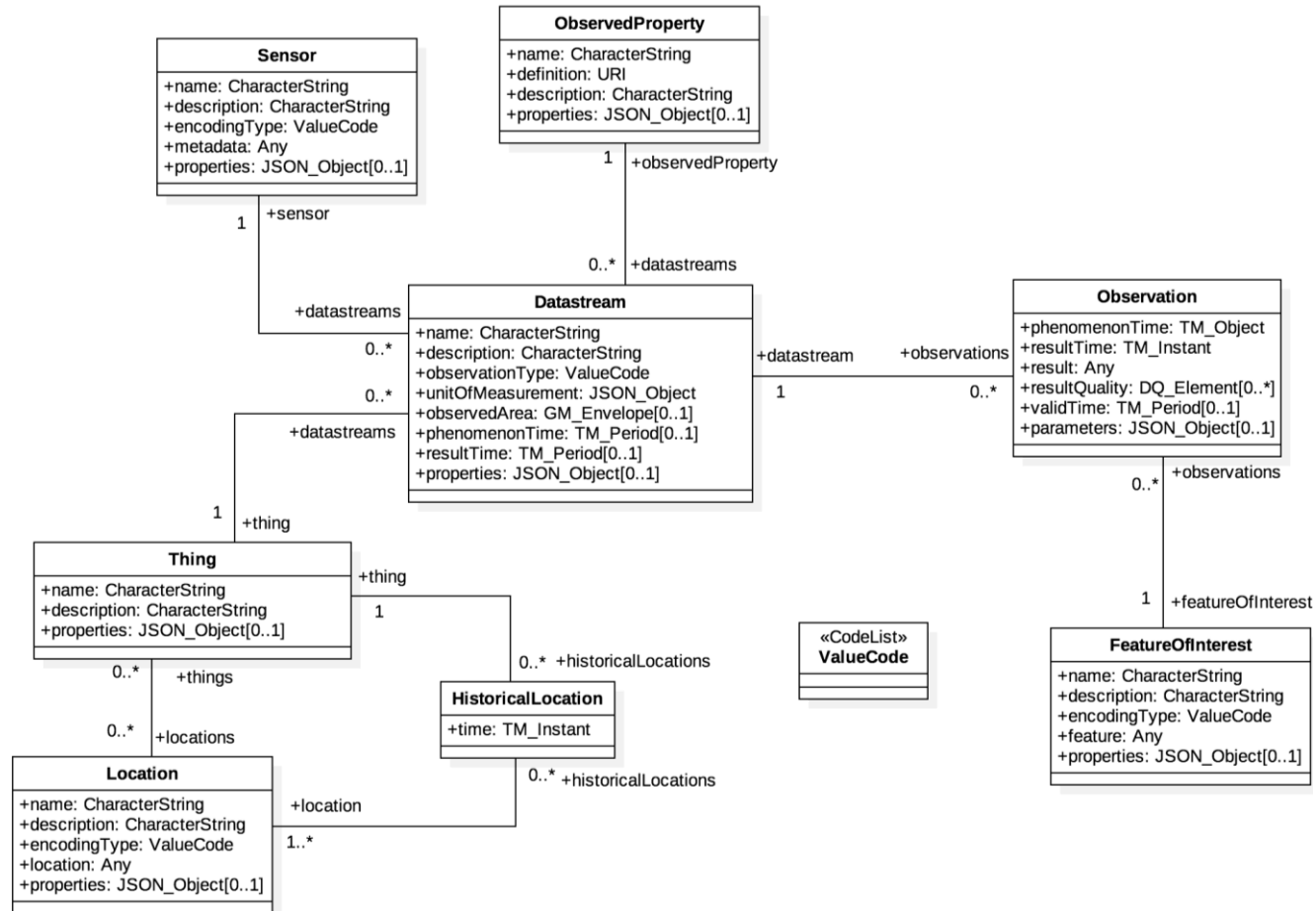3. **simplifies connections** between device-devices and device-applications.

OGC SensorThings API

Open, geospatial, unified solution for interconnecting Internet of Things (IoT) devices, data, and applications on the web.

- **Sensing part** → standard way to manage and retrieve observations and metadata from heterogeneous IoT sensor systems.

- **Tasking part** [Future work] → standard way to parameterize (tasking) IoT devices.

OGC SensorThings API

Open, geospatial, unified solution for interconnecting Internet of Things (IoT) devices, data, and applications on the web.

- **Sensing part** → standard way to manage and retrieve observations and metadata from heterogeneous IoT sensor systems.

- **Tasking part** [Future work] → standard way to parameterize (tasking) IoT devices.
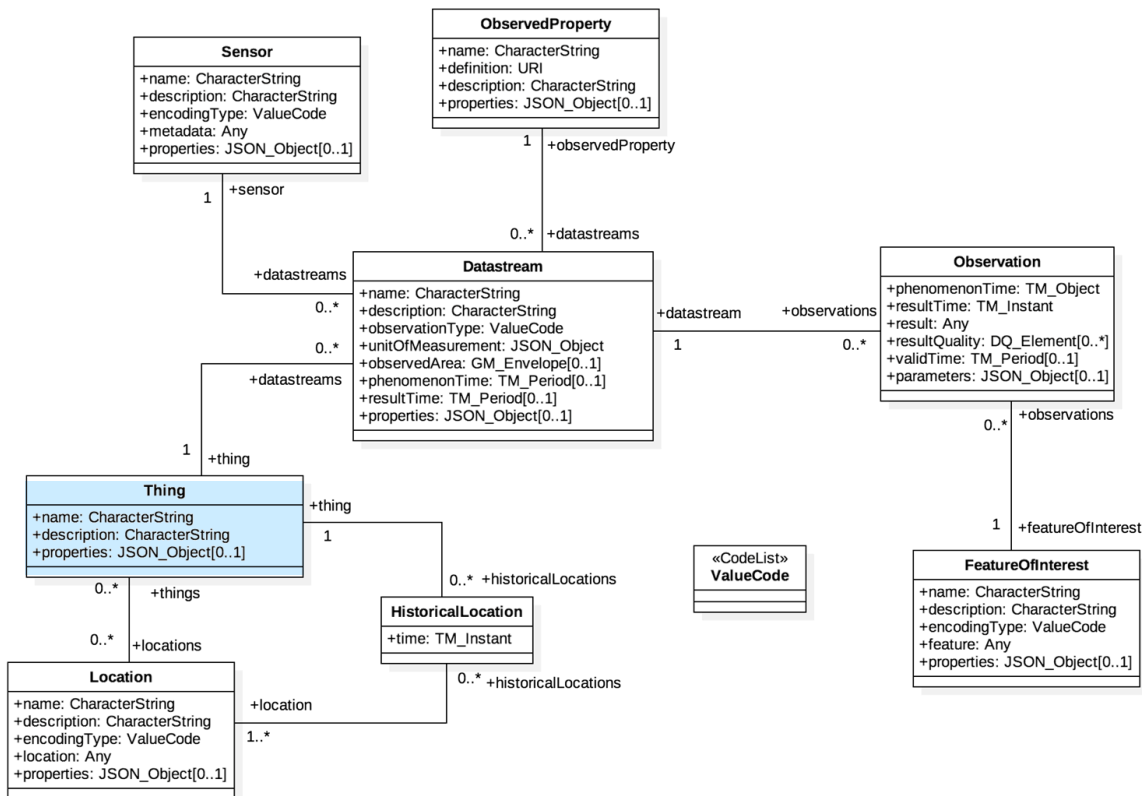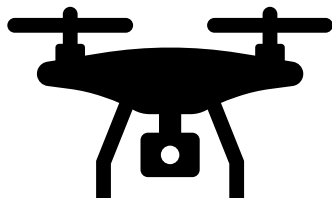
OGC SensorThings API Part1: Sensing Version 1.1

OGC SensorThings API Part1: Sensing Version 1.1

Thing: object of the physical world or of the information world that is capable of being identified and integrated into communication networks.

Example:
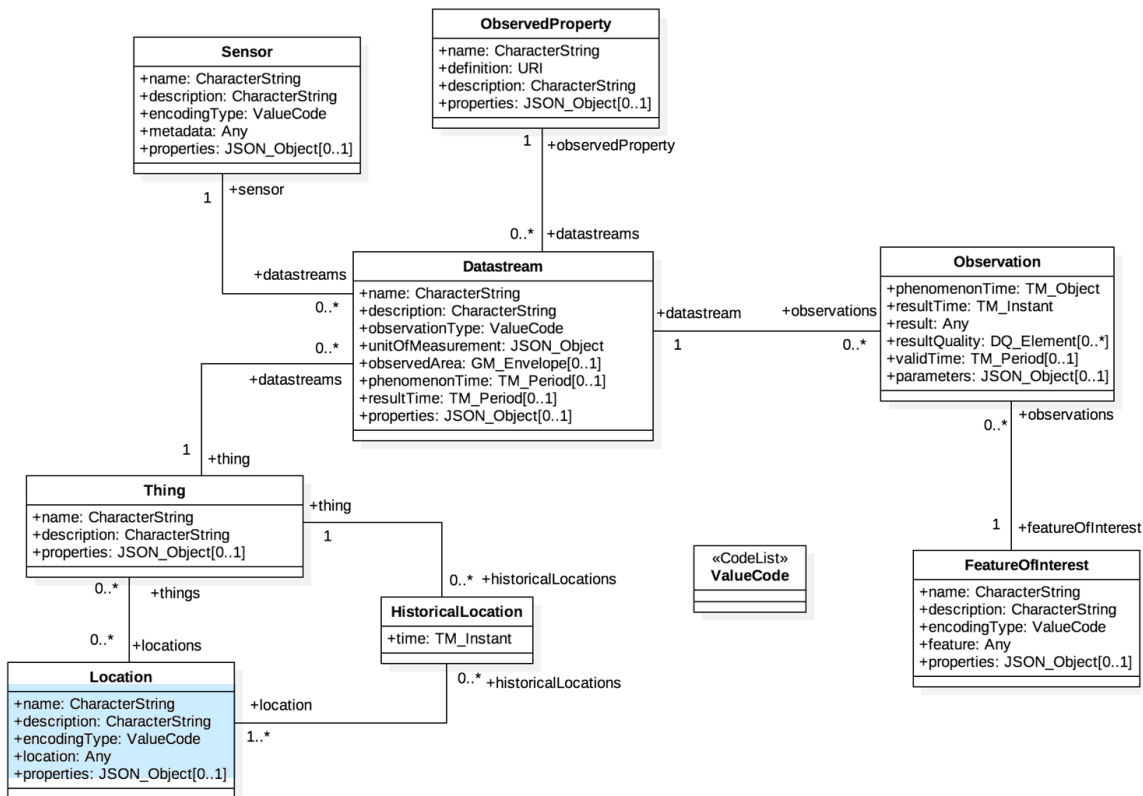**A drone surveys a relief along a route to retrieve its hyperspectral reflectance.**

Location: the last known position of the object.

Example:
A drone surveys a relief along a route to retrieve its hyperspectral reflectance.
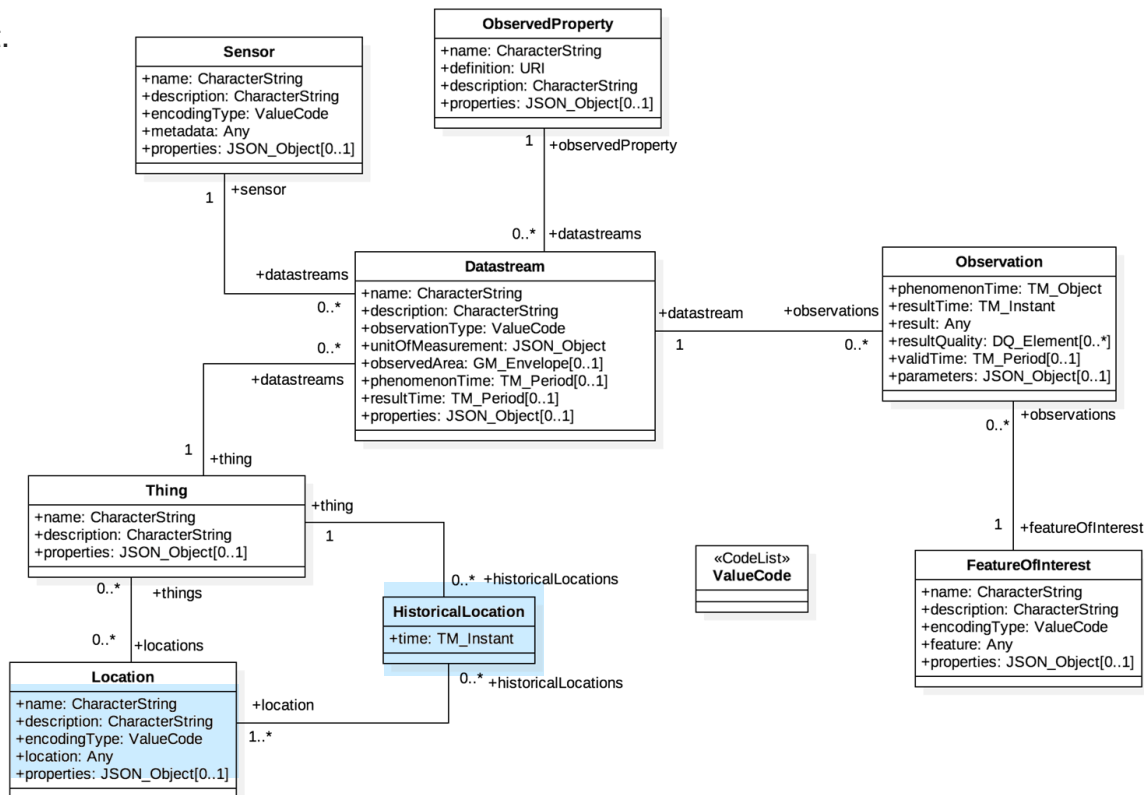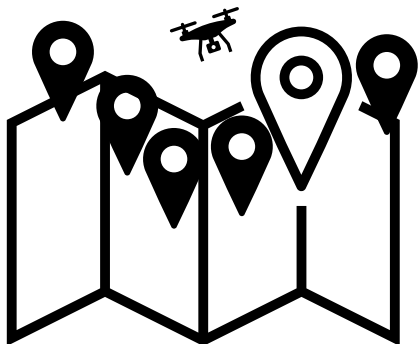
OGC SensorThings API Part1: Sensing Version 1.1

Location: the last known position of the object.

HistoricalLocation: the list of past known positions of the object.

Example:
A drone surveys a relief along a route to retrieve its hyperspectral reflectance.
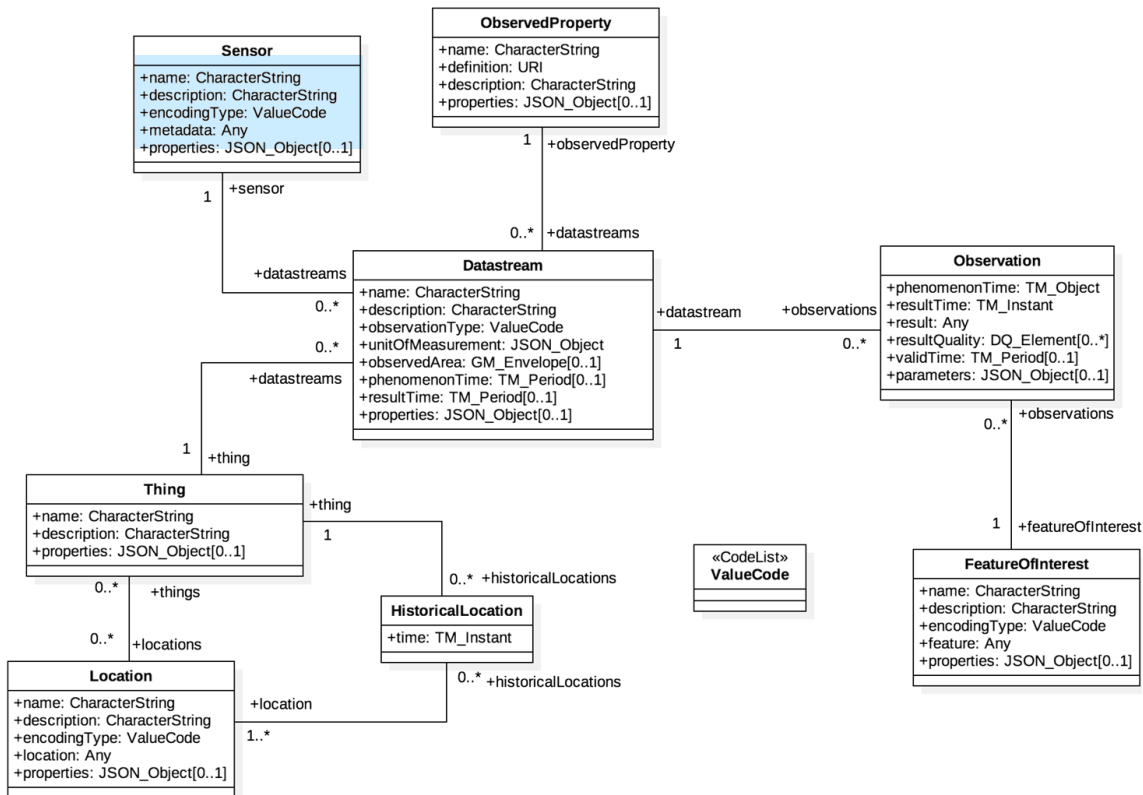
OGC SensorThings API Part1: Sensing Version 1.1

Sensor: device that observes a property or phenomenon with the aim of producing an estimate of the property's value (including human observers).

Example:
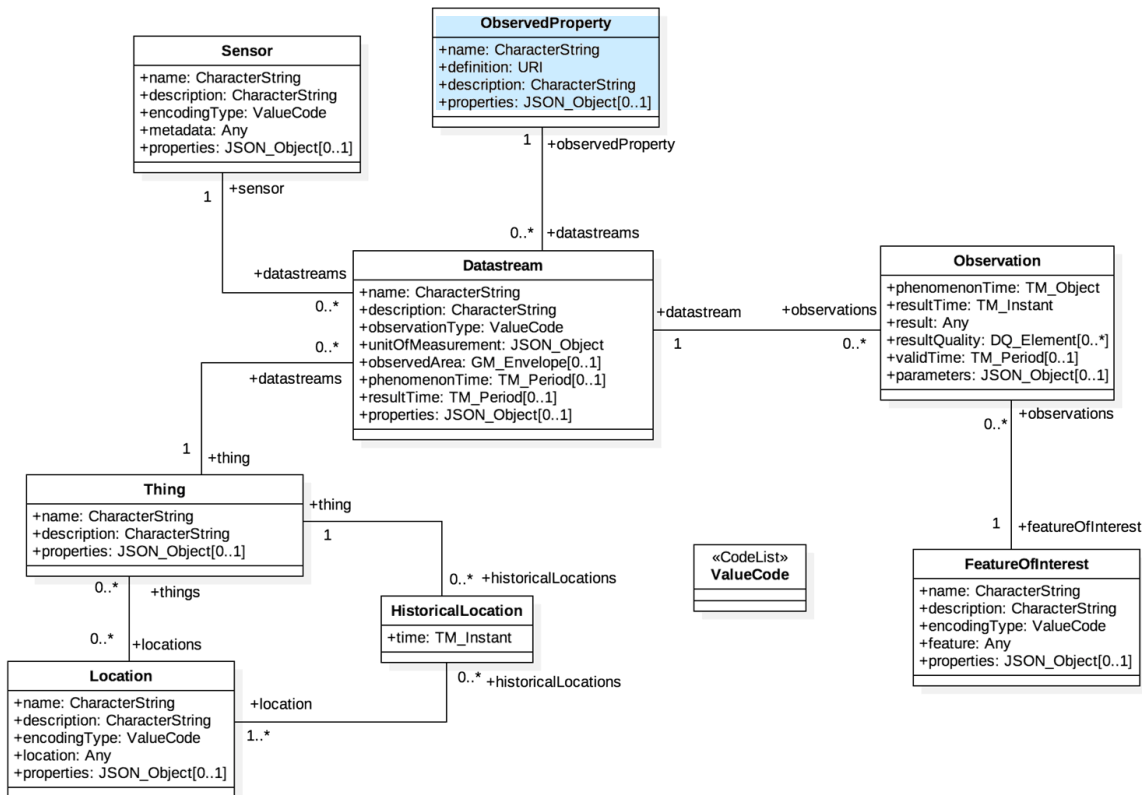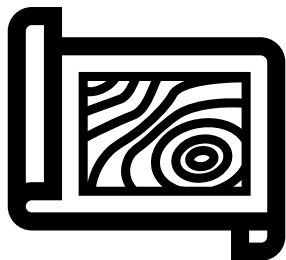**A drone surveys a relief along a route to retrieve its hyperspectral reflectance.**

ObservedProperty: the physical property of the feature that is being observed.

Example:
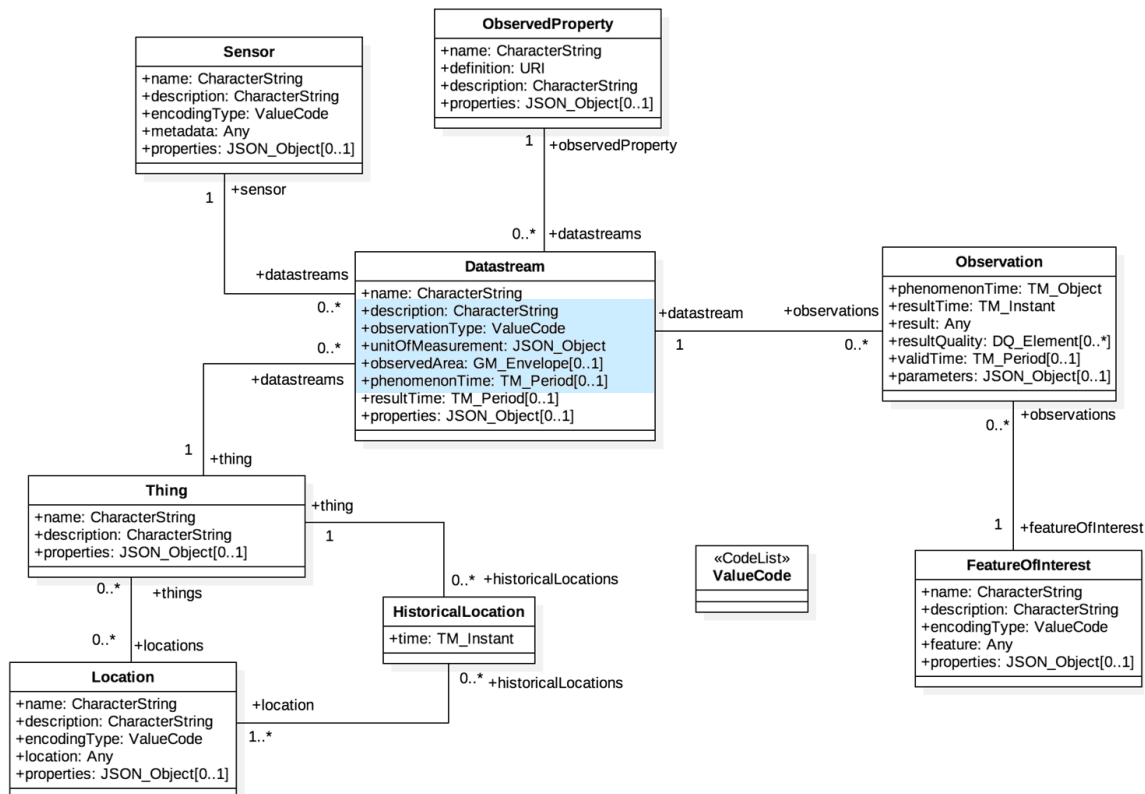**A drone surveys a relief along a route to retrieve its hyperspectral reflectance.**

OGC SensorThings API Part1: Sensing Version 1.1

deda.next
EDIAQI

Datastream: brings together a collection of Observations measuring the same Observed Property and produced by the same Sensor.

Example:
A drone surveys a relief along a route to retrieve its hyperspectral reflectance.

ObservedProperty
+name: CharacterString
+definition: URI
+description: CharacterString
+properties: JSON_Object[0..1]

Sensor
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+metadata: Any
+properties: JSON_Object[0..1]

1   +observedProperty

1   +sensor

0..*   +datastreams

+datastreams

0..*

0..*

+datastreams

Datastream
+name: CharacterString
+description: CharacterString
+observationType: ValueCode
+unitOfMeasurement: JSON_Object
+observedArea: GM_Envelope[0..1]
+phenomenonTime: TM_Period[0..1]
+resultTime: TM_Period[0..1]
+properties: JSON_Object[0..1]

+datastream   +observations

1   0..*

Observation
+phenomenonTime: TM_Object
+resultTime: TM_Instant
+result: Any
+resultQuality: DQ_Element[0..*]
+validTime: TM_Period[0..1]
+parameters: JSON_Object[0..1]

0..*   +observations

1   +thing

1   +featureOfInterest

Thing
+name: CharacterString
+description: CharacterString
+properties: JSON_Object[0..1]

+thing

1

0..*   +things

0..*   +locations

0..*   +historicalLocations

HistoricalLocation
+time: TM_Instant

0..*   +historicalLocations

«CodeList»
ValueCode

FeatureOfInterest
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+feature: Any
+properties: JSON_Object[0..1]

Location
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+location: Any
+properties: JSON_Object[0..1]
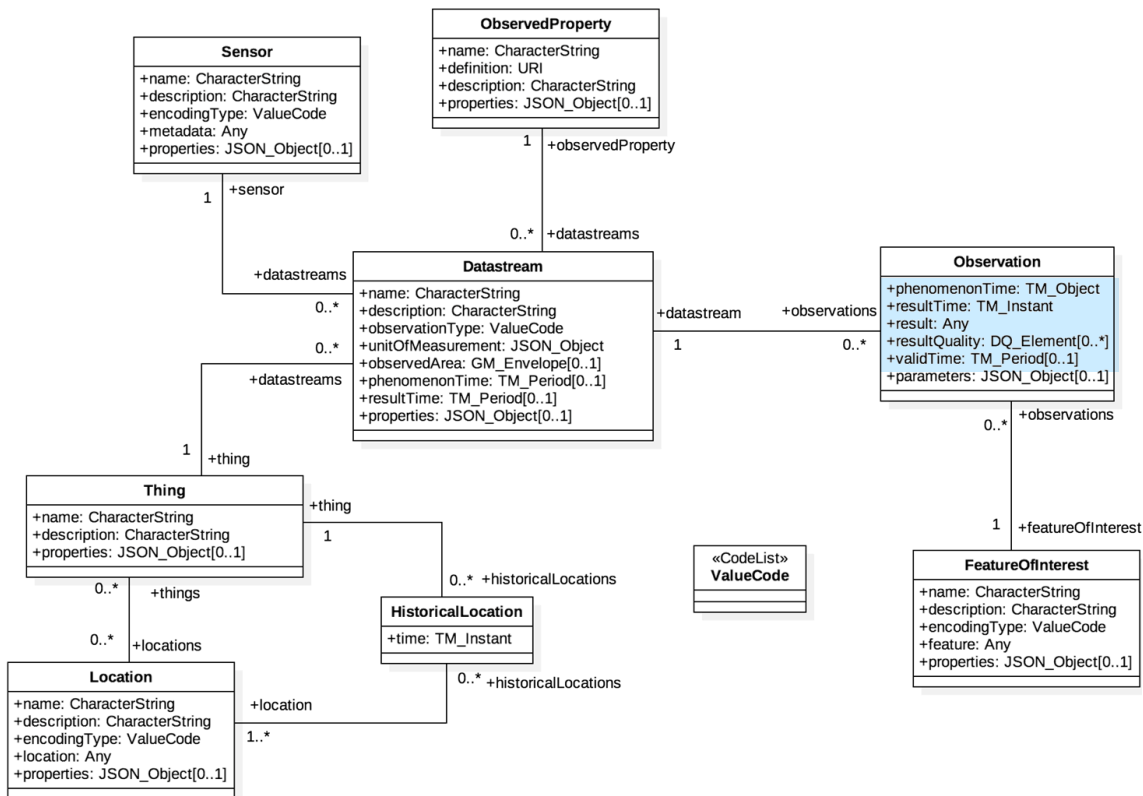
+location

1..*

OGC SensorThings API Part1: Sensing Version 1.1

Observation: An Observation is the act of measuring or otherwise determining the value of a property.

Example:
A drone surveys a relief along a route to retrieve its hyperspectral reflectance.

Sensor
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+metadata: Any
+properties: JSON_Object[0..1]

ObservedProperty
+name: CharacterString
+definition: URI
+description: CharacterString
+properties: JSON_Object[0..1]

1 +observedProperty

1 +sensor

0..* +datastreams

+datastreams

0..*

0..*

Datastream
+name: CharacterString
+description: CharacterString
+observationType: ValueCode
+unitOfMeasurement: JSON_Object
+observedArea: GM_Envelope[0..1]
+phenomenonTime: TM_Period[0..1]
+resultTime: TM_Period[0..1]
+properties: JSON_Object[0..1]

+datastream   +observations

1   0..*

Observation
+phenomenonTime: TM_Object
+resultTime: TM_Instant
+result: Any
+resultQuality: DQ_Element[0..*]
+validTime: TM_Period[0..1]
+parameters: JSON_Object[0..1]

+datastreams

1 +thing

0..* +observations

1 +featureOfInterest

Thing
+name: CharacterString
+description: CharacterString
+properties: JSON_Object[0..1]

+thing
1

«CodeList»
ValueCode

FeatureOfInterest
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+feature: Any
+properties: JSON_Object[0..1]

0..* +things

0..* +historicalLocations

0..*

0..* +locations

HistoricalLocation
+time: TM_Instant

Location
+name: CharacterString
+description: CharacterString
+encodingType: ValueCode
+location: Any
+properties: JSON_Object[0..1]

+location

1..*
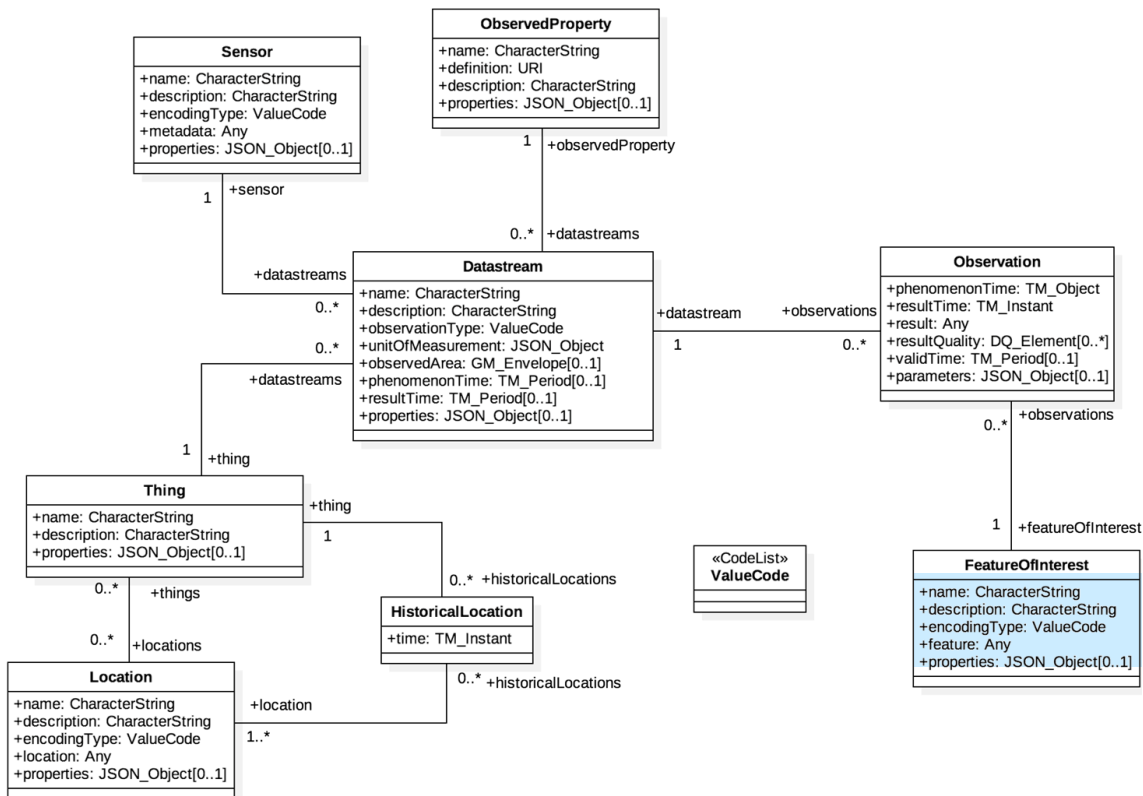
0..* +historicalLocations

OGC SensorThings API Part1: Sensing Version 1.1

FeatureOfInterest: An observation involves assigning a value to a phenomenon. The phenomenon is a property of a feature, the latter being the feature of the observational interest.

Example:
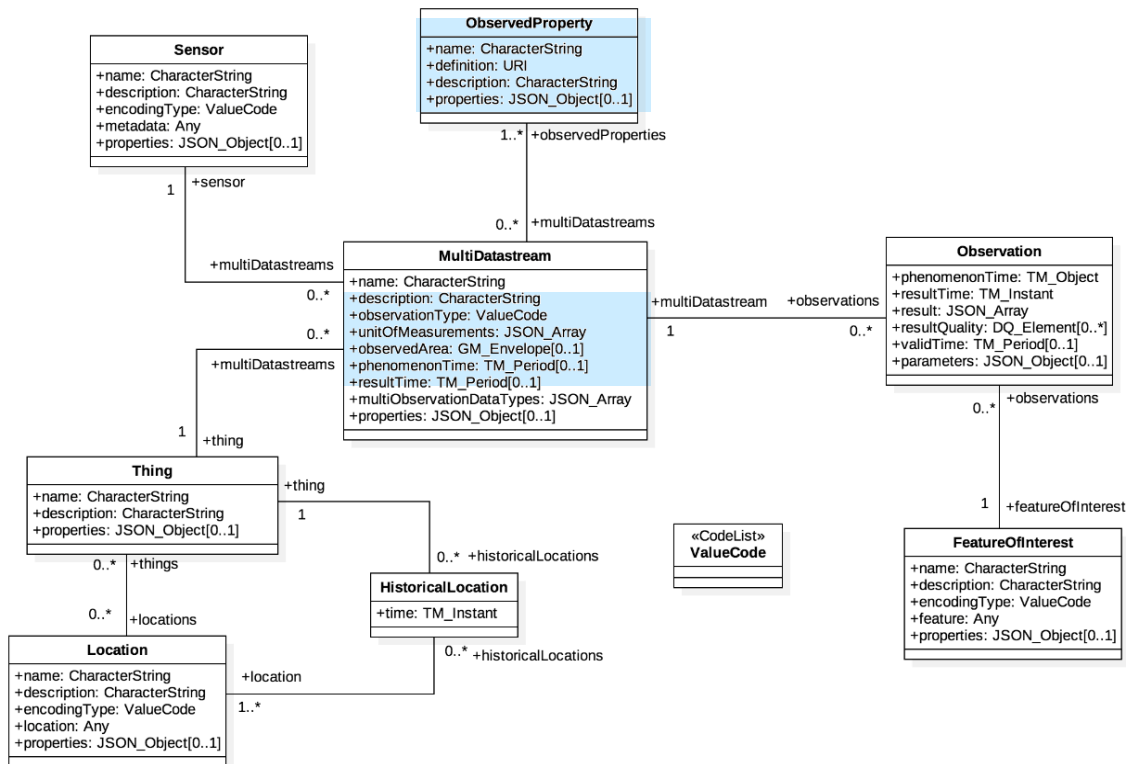**A drone surveys a relief along a route to retrieve its hyperspectral reflectance.**

MultiDatastream extension

A **MultiDatastream** is an extension to handle complex observations when the result is an array.

Example:
A drone surveys a relief along a route to retrieve its hyperspectral and radar reflectance.

OGC SensorThings API – Tailoring responses

- **$top**: specify the maximum number of objects to be returned. The usual default setting for $top is 100.

- **$skip**: used for paging, skip over the first n records and provide records from the n + 1 on.

- **$count**: return the total number of objects in the response. The usual default setting for $count is false.

- **$orderBy**: used to specify that the returned objects should be ordered by a specific attribute, either ascending or descending.

- **$select**: specify exactly which attributes are to be provided in the response.

- **$filter**: specify filters that control which entities are returned. See Filtering

- **$expand**: create a response returning multiple object types nested within each other.

OGC SensorThings API 1.1 Extension: STAplus 1.0

OGC SensorThings API: Conclusions
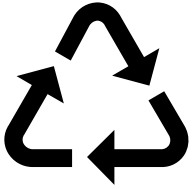
- **Flexible** data model → can be used easily for different cases
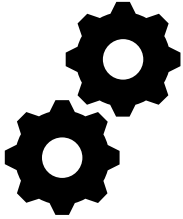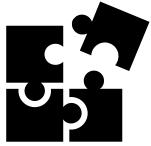
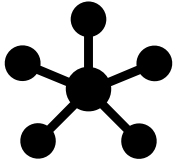- **Open** technology → easy interoperability, multiple implementations

- **International** Open Standard → based on consensus, a widely adopted solution
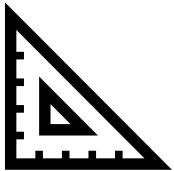
OGC SensorThings API: Conclusions

- **Lightweight** and **extensible**

- Built on both the **ISO** observation model and modern data exchange mechanisms

- Supports different **levels of detail** and different **quality** measures