
W3C Web of Things & OGC SensorThings API

How they can work together and benefit from each other

Michael Jacoby (Fraunhofer IOSB), Hylke van der Schaaf (Fraunhofer IOSB),
Josh Lieberman (OGC), Kathi Schleidt (DataCove e.U.)



Munich, 05.06.2019

AGENDA

- What is OGC SensorThings API (STA) and how does it work?
- How do WoT and STA relate?
- How can WoT benefit from STA?
- How can STA benefit from WoT?

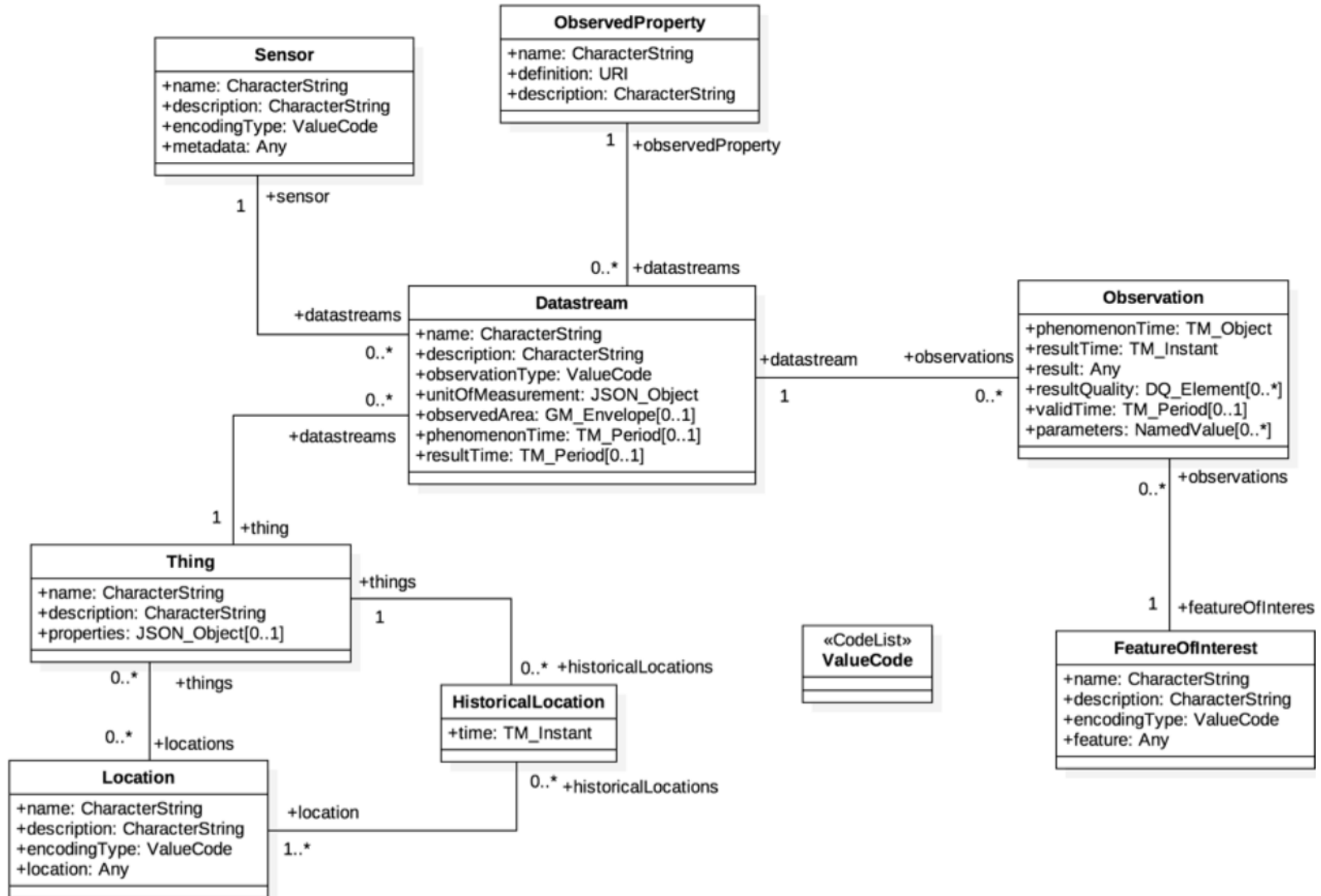
What is the OGC SensorThings API?



- Free & open web-based API for managing IoT devices
 - Focus on sensors and observations
 - Motivated by previous OGC standards like SWE, SOS, SPS, O&M
- Technical Aspects
 - HTTP/RESTful
 - MQTT
 - JSON
 - Adapting OData URL patterns and query options
- Relevance
 - (Open-source) implementations for server and client available
 - High visibility in environmental sensing and smart city communities
 - Many systems already deployed
 - Often with (tens of) millions of observations

<https://www.opengeospatial.org/standards/sensorthings>

STA Data Model



STA Basics – Accessing Entities

HTTP GET [http://example.org/SensorThingsService/v1.0/Things\(1\)](http://example.org/SensorThingsService/v1.0/Things(1))

Response

```
{
  "@iot.id" : 1,
  "name" : "Room S021",
  "description" : "This is room S021",
  "properties" : {
    "roomNumber" : "S021",
    "numberOfSeats" : "4"
  },
  "Locations@iot.navigationLink" : "Things(1)/Locations",
  "HistoricalLocations@iot.navigationLink" : "Things(1)/HistoricalLocations",
  "Datastreams@iot.navigationLink" : "Things(1)/Datastreams",
  "@iot.selfLink" : "/SensorThingsService/v1.0/Things(1)"
}
```

STA Basics – Using Links

HTTP GET [http://example.org/SensorThingsService/v1.0/Things\(17\)?
\\$select=@iot.id,description&
\\$expand=Datastreams\(\\$select=@iot.id,description\)](http://example.org/SensorThingsService/v1.0/Things(17)?$select=@iot.id,description&$expand=Datastreams($select=@iot.id,description))

Response

```
{
  "description" : "camping lantern",
  "@iot.id" : 17,
  "Datastreams" : [
    {
      "description" : "Temperature measurement",
      "@iot.id" : 19
    },
    {
      "description" : "Humidity measurement",
      "@iot.id" : 21
    }
  ]
}
```

STA Basics – Filtering

HTTP GET [http://example.org/SensorThingsService/v1.0/Observations?](http://example.org/SensorThingsService/v1.0/Observations?&filter=result%20gt%205)
`$filter=result gt 5`

Response:

```
{
  "@iot.count" : 8,
  "@iot.nextLink" : "/v1.0/Observations?$filter=result gt 5&$top=4&$skip=4",
  "value" : [
    {
      "phenomenonTime" : "2016-06-22T13:21:31.144Z",
      "resultTime" : null,
      "result" : 10,
      "@iot.id" : 34,
      "@iot.selfLink" : "/SensorThingsService/v1.0/Observations(34)"
    }, {
      ...
    }, {
      ...
    }, {
      ...
    }
  ]
}
```

How do WoT and STA relate?

- WoT
 - Meta-level/API description
 - Communication protocol-agnostic
 - Paradigm: state-based
- STA
 - Tailored to sensing & actuation domain
 - Concrete communication protocols
 - Paradigm: state-based (meta-data), observation-based
- → Rather complementary than competitive

How can WoT benefit from STA?

- STA as use case for TD
 - New requirements for TD, e.g.
 - Linking between (collections) of Things
 - Recursive datatypes
- Embrace (geospatial) nature of data
 - Geospatial information as part of WoT?
 - Pattern for historic values?
- Resource discovery and resource access
 - Querying data and metadata within same query (ThingDirectory)
 - Efficient resource access using \$expand (ThingDirectory, Scripting API)

STA Entity as WoT TD

```
{
  "@context": { "ex": "http://example.com/myModel/" },
  "name": "Room S021",
  "@type": ["Thing", "ex:Room"],
  "id": "http://example.com/STA/v1.0/Things(1)",
  "properties": {
    "temperature": {
      "@type": "ex:roomTemperature",
      "label": "room temperature in S021",
      "type": "number",
      "readOnly": true,
      "forms": [ {
        "href": "http://example.com/STA/v1.0/Observations?$filter=
          Datastream/Thing/id eq 1 and Datastream/ObservedProperty/name eq
            'temperature' & $orderBy=resultTime desc & $top=1 &
            $select=result",
        "mediaType": "application/json"
      } ]
    }
  }
}
```

How can STA benefit from WoT?

- Additional semantic
 - Use Semantic Web technologies
 - Domain-/Application-specific data models
- Interoperability and Federations
 - Expose Things across multiple STA servers in a unified way
 - Interoperability with other systems/devices

Discussion & Questions