New Features and Relevant Fixes in NGSI-LD v1.4.2

Presented by: Martin Bauer, Jason Fox, Giuseppe Tropea and Ken Zangelin

For: ETSI ISG CIM

22.12.2022
Overview

• New Features in NGSI-LD v1.4.2
  • Support for natural languages via LanguageProperty
  • Temporal Aggregation Functions

• Some Fixes in NGSI-LD v1.4.2 and their Rationale
  • Align Table 6.8.3.2-1 with clause 5.10.2-2 for query via attrs
  • Dot vs. comma in DateTime
  • id and type for JSON-LD compliance

[CIM 009v1.4.2] https://www.etsi.org/deliver/etsi_gs/CIM/001_099/009/01.04.02_60/gs_CIM009v010402p.pdf
New Features in NGSI-LD v1.4.2
(Natural) Language Properties in NGSI-LD

- Example – place with name in multiple languages

  ```json
  {
    "id": "urn:ngsi-ld:PointOfInterest:001",
    "type": "PointOfInterest",
    "exonym": {
      "type": "LanguageProperty",
      "languageMap": {
        "fr": "Grand Place",
        "nl": "Grote Markt"
      }
    }
  }
  ```

- Languages are specified using BCP47 language codes, e.g. “fr”, “en” or “en-US”

- Language filter is added as parameter, e.g. `lang="fr"`, in this case the Language Property is converted to a regular Property using the specified language in the result:

  ```json
  {
    "id": "urn:ngsi-ld:PointOfInterest:001",
    "type": "PointOfInterest",
    "exonym": {
      "type": "Property",
      "value": "Grand Place"
    }
  }
  ```

- In a filter query, either an explicit language to match is specified, e.g. `exonym[fr]= "Grand Place"` or any language can match, e.g. `exonym[*]= "Grand Place"`

[CIM 009v1.4.2] Clause 4.2.2, 4.6.5, 4.9, 4.15, 5.2.23, 5.2.32, 5.7.1, 5.7.2, 5.7.3, 5.7.4, 6.4.3.2, 6.5.3.1, 6.18.3.2, 6.19.3.1, Annex B, C.2.2
Suggested Best Practices for Internationalization

- Use standard vocabulary from schema.org to identify the language and sameAs relationship to refer to the same entity, e.g. provided in a different language:
  - "inLanguage": "http://schema.org/inLangauge"
  - "sameAs": http://schema.org/sameAs

```json
{
  "type": "Event",
  "id": "urn:ngsi-ld:Event:bonjourLeMonde",
  "name": {
    "type": "Property",
    "value": "Bonjour le Monde"
  },
  "sameAs": [
    {
      "type": "Relationship",
      "datasetId": "urn:ngsi-ld:Relationship:1",
      "object": "urn:ngsi-ld:Event:helloWorld",
      "inLanguage": {
        "type": "Property",
        "value": "en"
      }
    },
    {
      "type": "Relationship",
      "object": "urn:ngsi-ld:Event:halloWelt",
      "inLanguage": {
        "type": "Property",
        "value": "de"
      }
    }
  ]
}
```
Aggregation Operators for Temporal Query Language - Operations

- Allows aggregation of temporal information

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning*</th>
</tr>
</thead>
<tbody>
<tr>
<td>totalCount</td>
<td>Calculate the number of times the value has been updated inside the period</td>
</tr>
<tr>
<td>distinctCount</td>
<td>Calculate the count of distinct values inside the period</td>
</tr>
<tr>
<td>sum</td>
<td>Calculate the sum of the values inside the period</td>
</tr>
<tr>
<td>avg</td>
<td>Calculate the average of the values inside the period</td>
</tr>
<tr>
<td>min</td>
<td>Calculate the minimum value inside the period</td>
</tr>
<tr>
<td>max</td>
<td>Calculate the maximum value inside the period</td>
</tr>
<tr>
<td>stddev</td>
<td>Calculate the standard deviation of the values inside the period</td>
</tr>
<tr>
<td>sumsq</td>
<td>Calculate the sum of the square of the values inside the period</td>
</tr>
</tbody>
</table>

*Meaning for numbers, some operators are also available for other datatypes, see specification

[CIM 009v1.4.2] Clauses 4.5.19, 5.7.3, 5.7.4, 6.3.12, 6.18.3.2, 6.19.3.1, Annex B, C.5.13
Aggregation Operators for Temporal Query Language – Aggregation Period

- The duration is expressed using the **ISO 8601 Duration** Representation
- The duration shall be a string in the format
  - or \( P[n]W \) (Weeks)

where \( [n] \) is replaced by the value for each of the date and time elements that follow

- Example: "\( P3Y6M4DT12H30M5S \)" represents a duration of “three years, six months, four days, twelve hours, thirty minutes, and five seconds”.

- Date and time elements including their designator may be omitted if their value is zero, e.g. \( PT4M \) represents a duration of “4 minutes.”

- Lower-order elements may be omitted for reduced precision

- A duration of 0 seconds (e.g., expressed as PT0S or P0D) is valid and is interpreted as a duration spanning the whole time range specified by the temporal query.
Aggregation Operators for Temporal Query Language - Example

GET /ngsi-ld/v1/temporal/entities/?type=Vehicle&q=brandName!="Mercedes"&attrs=speed&timerel=between&timeAt=2018-08-01T12:00:00Z&endTimeAt=2018-08-01T13:00:00Z&aggrMethods=max,avg&aggrPeriodDuration=PT4M&options=aggregatedValues
Accept: application/json
Link: <http://example.org/ngsi-ld/latest/aggregatedContext.jsonld>; rel="http://www.w3.org/ns/json-ld#context"; type="application/ld+json"

200 OK
Content-Type: application/json

[
  {
    "id": "urn:ngsi-ld:Vehicle:B9211",
    "type": "Vehicle",
    "speed": {
      "type": "Property",
      "max": [
        [120,"2018-08-01T12:00:00Z","2018-08-01T12:04:00Z"],
        [100,"2018-08-01T12:04:00Z","2018-08-01T12:08:00Z"]
      ]
    }
  }
]
Some Fixes in NGSI-LD v1.4.2 and their Rationale
Important Fixes

Align Table 6.8.3.2-1 with clause 5.10.2-2 for query via attrs

In Table 6.8.3.2-1 the required presence of type or attribute (possibly also geoquery or query) was added to reflect the behaviour described in the referenced clause 5.10.2

[Cl 6.8.3.2]

Dot vs. comma in DateTime

We followed ISO8601 that prefers comma for separating decimal parts of timestamps – unfortunately the world primarily uses dots and this creates problems. Change: allow both for now, but dots are now the preferred representation

[Cl 4.6.3]

id and type for JSON-LD compliance

NGSI-LD uses id and type, which are mapped to @id and @type generally used in JSON-LD. NGSI-LD now also allows the direct use of @id and @type in user-provided NGSI-LD content.

[Cl 4.4]