**Schema Options**

**[1] Look at metadata standards :** [**www.dcc.ac.uk**](http://www.dcc.ac.uk/)

**[2] Promising: “Observations and Measurements” standard:** <https://www.dcc.ac.uk/resources/metadata-standards/observations-and-measurements>   
 <https://portal.ogc.org/files/?artifact_id=64910>

**[3] Also promising: SenML (lightweight Sensor Measurement Lists):** <https://datatracker.ietf.org/doc/html/rfc8428>

**[4] List of metadata standards:** <https://www.dcc.ac.uk/guidance/standards/metadata/list> **[5] Research paper on schemata construction with sensor network example (page 4)** <https://research-repository.st-andrews.ac.uk/bitstream/handle/10023/15773/sdw.pdf;jsessionid=EBFC8FDA39EDBB8BFAE8518115BD04D6?sequence=1>

**[6] Sphere paper** [1805.11907v2.pdf](https://discovery-pp.ucl.ac.uk/id/eprint/10111763/1/1805.11907v2.pdf)

**[7] Structuring JSON schema** [2872427.2883029](https://dl.acm.org/doi/pdf/10.1145/2872427.2883029)

**[8] JSON Schema Draft-07** <https://json-schema.org/draft-07/draft-handrews-json-schema-01#rfc.section.4.3>

Accelerometer 3 axis (3 floats, 1 per axis) - in SI unit

* { ’n’: ’ ACCEL ’, ’t’: 0.32 , ’v’: [ -0.096 , -0.96, -0.192 ] }, [[6] p.g. 12, wearable message example]
* + WEARABLE ADV – internal only, contains wearable acceleration data [[6] p.g. 7, SPG-2 forwarding gateway nodes]
* ACCEL – acceleration data – v – the x, y, z coordinates of the acceleration data (always in this order) – t – the timestamp offset (in seconds) of this data relative to the timestamp of the document. [[6] p.g.13]
* Unit:
  + “if the record has no unit, the base unit is used as the unit” [3]
  + “Acceleration is quantified in the [SI](https://en.wikipedia.org/wiki/International_System_of_Units) unit [metres per second per second](https://en.wikipedia.org/wiki/Metre_per_second_squared) (m/s2)” [Wiki ‘Accelerometer’]

Gyroscope 3 axis (3 floats, 1 per axis)

* Measure the angular velocity around the x, y and z axis.
* { ’n’: ’ GYRO ’, ’t’: 0.32 , ’v’: [ -0.096 , -0.96, -0.192 ] },
* "gyroscope\_x": -0.02,  
  "gyroscope\_y": 0.15,  
  "gyroscope\_z": -0.08,
* Unit:
  + revolutions per second (RPS) or degrees per second (°/s)
  + “SI units of angular velocity are dimensionally equivalent to [reciprocal seconds](https://en.wikipedia.org/wiki/Reciprocal_seconds), s−1, although rad/s is preferable to avoid confusion with **rotation velocity** in units of [hertz](https://en.wikipedia.org/wiki/Hertz) (also equivalent to s−1)” [Wiki ‘Angular Velocity’]

Timestep (unitless – ticks – incrementing number)

* ’ts’: 80569776 , [[6] p.g. 12/all example JSON]

Temperature (deg C)

* Surrounding temperature

Pressure (hPA)

* BMP PRES – BMP sensor’s air pressure value, in hPa. [[6] p.g. 9, value of the pressure in the environmental nodes]

Battery (voltage)

* { ’ \_id ’: ObjectId (’5 ab989d57ea12b8f29eca79f ’), ’ tso ’: 1520032700.7959356 , ’e’: [{ ’n’: ’ BATMON\_VOLT ’, ’v’: 3.333 }, { ’n’: ’ UPTIME ’, ’v’: 4937677 }], ’ uid ’: ’a0:e6:f8:00:ff:c2’, ’ts’: 207618107 , ’bt’: ’ 2018 -03 -27 T00 :01: 21.865 Z’, ’ hid ’: ’ 0000 ’, ’ Date ’: ’ 2018 -03 -27 T00 :01: 22.325 Z’ } [pg. 21]
* BATMON VOLT – battery voltage of the device in volts [[6] p.g. 21]
* CPUVDD – CPU voltage in volts

SOC temperature (deg C)

* Temperature of the chip in the device
* CPUTEMP – CPU temperature in oC. [[6] p.g. 21]

Any extra features/products of processing on wearable (TBC)

* RSSI information (Gateway)
  + ’gw’: [{ ’ uid ’: ’ fd00 :: 212 :4 b00 :0: ff05 ’, ’ rssi ’: -89, ’mc’: 171761 , ’ts’: 80569981 }, [old report, p.g. 12]
  + { ’n’: ’ RSSI ’, ’v’: -30 }, [[6] pg. 15]
* UID – ID of the device.
* Location (of the wearable) (Gateway)
  + {’n’: ’ LOCATION ’, ’v’: ’ sphere office 1’}, [[6] p.g. 15]
* Wearable error messages
  + typedef enum { ERROR\_NO\_ERROR =0 , // Everything OK ERROR\_AXL0\_DEAD , // Accelerometer 0 not responding ERROR\_AXL0\_FIFO , // Accelerometer 0 unexpected number of samples in FIFO ERROR\_AXL0\_FIFO\_FULL , // Accelerometer 0 has full FIFO ERROR\_AXL1\_DEAD , // Accelerometer 1 not responding ERROR\_AXL1\_FIFO , // Accelerometer 1 unexpected number of samples in FIFO ERROR\_SPI0\_FAIL , // Failed to initialize SPI0 ERROR\_SPI1\_FAIL , // Failed to initialize SPI1 ERROR\_FLASH , // Failed to initialize external flash memory ERROR\_ADV\_CONFIG , // Failed to configure accelerometer ERROR\_CRYPTO\_INIT , // Failed to initialize crypto engine ERROR\_CRYPTO\_OP // Failed to encrypt or decrypt } ERROR\_TYPE ; // Wearable Error Type [[6] p.g. 22]

Using JSON Schema draft-07:

- JSON file to describe the structure/properties of JSON packets

- Generic example (schema + data packet):

==============================================================

{

"$schema": "http://json-schema.org/draft-07/schema#",

"title": "Sensor Data Packet",

"type": "object",

"properties": {

"sensorId": { "type": "string" },

"timestamp": { "type": "string", "format": "date-time" },

"measurements": {

"type": "array",

"items": {

"type": "object",

"properties": {

"property": { "type": "string" },

"value": { "type": ["number", "string", "boolean"] },

"unit": { "type": "string" }

},

"required": ["property","value","unit"]

}

},

"location": {

"type": "object",

"properties": {

"lat": { "type": "number", "minimum": -90, "maximum": 90 },

"lon": { "type": "number", "minimum": -180,"maximum": 180 }

},

"required": ["lat","lon"]

},

"metadata": {

"type": "object",

"properties": {

"firmwareVersion": { "type": "string" },

"battery": { "type": "number" }

}

}

},

"required": ["sensorId","timestamp","measurements"]

}

==============================================================

{

"sensorId": "urn:dev:eui-70b3d57ed00001a2",

"timestamp": "2025-06-20T10:00:00Z",

"measurements": [

{ "property": "temperature", "value": 22.7, "unit": "°C" },

{ "property": "humidity", "value": 48.3, "unit": "%" }

],

"location": { "lat": 51.5074, "lon": -0.1278 },

"metadata": {

"firmwareVersion": "1.2.3",

"battery": 87

}

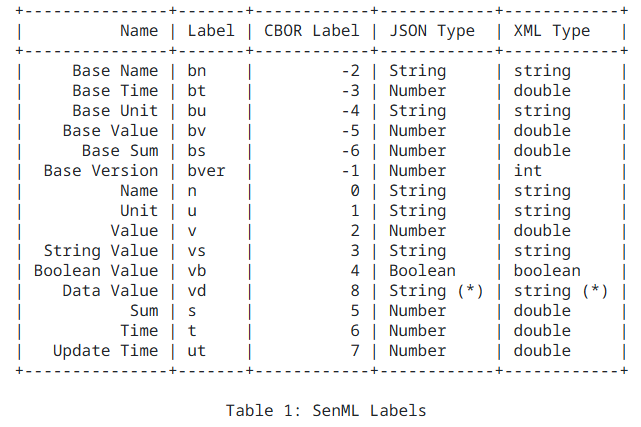
}

==============================================================

Implementing SenML:

- Uses base values to eliminate repeated information  
 - e.g. “bn” (base name) : “12638907fdsh67” ; following packets use “n” : “” to inherit the base name as a prefix. Following packets could also use “n”: “1” to append 1 to the base name.

- “bn” is base name; “bt” is base time; “bu” is base units; “bv” is base value; “bs” is base sum; “bver” is base version

 - Uses UTF-8 JSON only – good for compatibility

**Type 0 Wearable Device Schema and Example Packet**

<https://github.com/talliskinrade/torus_wearable_packet.git>