|  |  |
| --- | --- |
| Project | **Specification of Sensor Interface for Cyber and Physical World**<https://sagroups.ieee.org/2888/ > |
| Title | **Root schema structure using JSON of sensor data formats for cyber and physical world interface** |
| DCN | **2888-20-0008-01-0001** |
| Date Submitted | **Feb. 23, 2020**  |
| Source(s) | Sang-Kyun Kim, goldmunt@gmail.com (Myongji University)Kyoungro Yoon, yoonk@konkuk.ac.kr (Konkuk University)Sangkwon Jeong, ceo@joyfun.kr (Joyfun) |
| Re: |  |
| Abstract | This contribution illustrates the basic JSON schema structure for representing sensor information in the physical world in a standardized data format. Sensors in the physical world transform the detected data into a JSON instance that follows the JSON schema presented by this contribution and forwards it to the cyber world. Cyber world checks the adequacy of the sensor data by checking whether the sensor data in JSON format conforms to the standardized JSON schema. |
| Purpose | To start discussion on purpose of the standard |
| Notice | This document has been prepared to assist the IEEE 2888 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that IEEE 3079 may make this contribution public. |
| Patent Policy | The contributor is familiar with IEEE patent policy, as stated in [Section 6 of the IEEE-SA Standards Board bylaws](http://standards.ieee.org/guides/opman/sect6.html#6.3) <[http://standards.ieee.org/guides/bylaws/sect6-7.html#6](http://127.0.0.1:4664/cache?event_id=757737&schema_id=1&s=5X0vID10lu_E6yrIkWkNd4Wz2H8&q=hancock)> and in *Understanding Patent Issues During IEEE Standards Development* <http://standards.ieee.org/board/pat/faq.pdf> |

# Introduction

This contribution illustrates the basic JSON schema structure for representing sensor information in the physical world in a standardized data format. Sensors in the physical world transform the detected data into a JSON instance that follows the JSON schema presented by this contribution and forwards it to the cyber world. Cyber world checks the adequacy of the sensor data by checking whether the sensor data in JSON format conforms to the standardized JSON schema.

Section 2 describes the sensor data base schema that can include each sensor data schema, and section 3 concludes.

# Base schema structure for sensor data

{

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

  "title": "Sensor data",

  "description": "Schema for sensor data",

  "type": "object",

  "properties": {

    "timeStamp": {"type": "datetime"},

    "sensedInfoBaseAttributes": {

      "$ref": "#/definitions/sensedInfoBaseAttributes"

    },

    "microphoneSensorType": {

      "$ref": "#/definitions/microphoneSensorType"

    },

    "cameraSensorType": {

      "$ref": "#/definitions/cameraSensorType"

    },

"orientationSensorType": {

      "$ref": "#/definitions/orientationSensorType"

    },

    "temperatureSensorType": {

      "$ref": "#/definitions/temperatureSensorType"

    },

    "bloodPressureSensorType": {

      "$ref": "#/definitions/bloodPressureSensorType"

    }

  },

  "additionalProperties": false,

  "required": [

    "sensedInfoBaseAttributes"

  ],

  "minProperties": 2,

  "maxProperties": 3,

  "definitions": {

    "sensedInfoBaseAttributes": {

      "additionalProperties": false,

      "type": "object",

      "properties": {

        "id": {"type": "string"},

        "sensorIdRef": {"type": "string"},

        "linkedList": {"type": "string"},

        "groupID": {"type": "string"},

        "activate": {"type": "boolean"},

        "priority": {"type": "integer","minimum": 0}

      }

    },

"microphoneSensorType": {

 . . .

    },

 "cameraSensorType": {

 . . .

    },

 "orientationSensorType": {

 . . .

    },

 "temperatureSensorType": {

 . . .

    },

 "bloodPressureSensorType": {

 . . .

    },

}

}

figure 1. Base schema structure in JASON for sensor data

Figure 1 illustrates the basic structure of the JSON schema for representing sensor data coming from the physical world. It includes basic information and attributes that all sensor data should have in common and data format schema for each sensor that can express the information acquired by each sensor.

## Root properties

{

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

  "title": "Sensor data",

  "description": "Schema for sensor data",

  "type": "object",

  "properties": {

    "timeStamp": {"type": "datetime"},

    "sensedInfoBaseAttributes": {

      "$ref": "#/definitions/sensedInfoBaseAttributes"

    },

    "microphoneSensorType": {

      "$ref": "#/definitions/microphoneSensorType"

    },

    "cameraSensorType": {

      "$ref": "#/definitions/cameraSensorType"

    },

"orientationSensorType": {

      "$ref": "#/definitions/orientationSensorType"

    },

    "temperatureSensorType": {

      "$ref": "#/definitions/temperatureSensorType"

    },

    "bloodPressureSensorType": {

      "$ref": "#/definitions/bloodPressureSensorType"

    }

  },

  "additionalProperties": false,

  "required": [

    "sensedInfoBaseAttributes"

],

  "minProperties": 2,

  "maxProperties": 3,

figure 2. Root properties

Figure 2 shows the most basic properties for representing sensors and sensor information as the root properties of the sensor data schema.

The root property consists of timeStamp, sensedInfoBaseAttributes, and properties that can refer to the data type of each sensor data.

The timeStamp is a property that represents the time when sensor data is acquired. It is expressed in “dd/mm/yyyyhh:mm:ss” format.

The sensedInfoBaseAttributes is a property that represents the information obtained from the sensor and the basic attributes of the sensor.

The remaining properties (e.g., microphoneSensorType, cameraSensorType), except for timeStamp and sensedInfoBaseAttributes, refer to data types defined outside of the root property and represent the information obtained by each sensor.

The value of additionalProperties is set to false so that data not defined in the schema cannot be used.

The sensedInfoBaseAttributes is a required property and must be used when creating JSON instance.

In addition, the minProperties is set to 2 and the maxProperties is set to 3 so that it includes at least two properties including the sensedInfoBaseAttributes and data of a specific sensor or up to three properties including the timeStamp of input data.

## SensedInfoBaseAttributes

"sensedInfoBaseAttributes": {

      "additionalProperties": false,

      "type": "object",

      "properties": {

        "id": {"type": "string"},

        "sensorIdRef": {"type": "string"},

        "linkedList": {"type": "string"},

        "groupID": {"type": "string"},

        "activate": {"type": "boolean"},

        "priority": {"type": "integer","minimum": 0}

      }

    },

figure 3. sensedInfoBaseAttributes

Figure 3 shows an example of the data format of sensedInfoBaseAttributes that defines the basic attributes of sensor data.

The sensedInfoBaseAttributes consists of id, sensorIdRef, linkedList, groupID, activate, and priority.

The id is a unique identifier of the sensor data that allows you to identify each sensor data.

The sensorIdRef represents the unique id of the input sensor device.

The linkedList represents a sensorIdRef for the next sensor when a structure in which several sensors are connected.

The groupID is an identifier for a set of sensors with a multi-sensor structure.

The activate describes whether the sensor is activated. A value of "true" means the sensor is active and "false" means the sensor is inactive.

The priority may determine the priority when sensor data is acquired from several sensors at the same time. 1 is the highest priority and higher number means lower priority. If sensor data with high priority and sensor data with low priority are acquired and transmitted at the same time, sensor data with low priority may be lost.

## Placeholder for Individual sensor data types

"microphoneSensorType": {

 . . .

    },

 "cameraSensorType": {

 . . .

    },

 "orientationSensorType": {

 . . .

    },

 "temperatureSensorType": {

 . . .

    },

 "bloodPressureSensorType": {

 . . .

    },

figure 4. Placeholder for individual sensor data types

Figure 4 shows an example of a placeholder for each sensor data type. This section describes the data types of the sensors declared in the root properties. Each sensor type contains the information needed to represent the sensor data between braces.

# Conclusion

It is recommended that the JSON-based sensor data base schema proposed in this contribution be adopted as the basic structure of the sensor data schema of IEEE 2888.1.