|  |  |
| --- | --- |
| Project | **Specification of Sensor Interface for Cyber and Physical World**  <<https://sagroups.ieee.org/2888/> **>** |
| Title | **Trakers for a large space** |
| DCN | **2888-21-0038-00-0001** |
| Date Submitted | **June 27th, 2021** |
| Source(s) | Sang-Kyun Kim, [goldmunt@gmail.com](mailto:goldmunt@gmail.com) (Myongji University)  Min Hyuk Jeong, [jmh8900@gmail.com](mailto:jmh8900@gmail.com) (Myongji University) |
| Re: |  |
| Abstract | This contribution illustrates the JSON schema structure for representing tracking sensor data in the physical world in a standardized data format. The semantics and examples of the tracking sensor information are presented. |
| 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 2888 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 JSON schema structure for representing tracking sensor data in the physical world in a standardized data format. The semantics and examples of the tracking sensor information are presented.

# Tracking sensor data

## Message time type

### General

This subclause specifies a sensor data type, which describes values related to time.

### Syntax

|  |
| --- |
| "messageTimeType": {  "type": "object",  "properties": {  "messageTimeSec": {  "type": "number",  },  "messageTimeFracSec": {  "type": "number",  },  },  "additionalProperties": false  "maxProperties": 1  }, |

### Semantics

Semantics of the messageTimeType:

|  |  |
| --- | --- |
| Name | Definition |
| messageTimeType | Tool for describing sensor information acquisition time. The time is defined with either messageTimeSec or messageTimeFracSec. |
| messageTimeSec | Describes the sensor information acquisition time with an elapsed time(second) from 01.01.1970 |
| messageTimeFracSec | Describes the sensor information acquisition time with an elapsed time(millisecond) from 01.01.1970 |

### Examples

This example indicates that the sensor information was acquired in 1624695473 seconds later from 01.01.1970. In general time, it is Saturday, Jun 26th, 2021, 17:17:53 UTC+0900.

|  |
| --- |
| {  “messageTimeType”: {  “messageTimeSec”: 1624695473  }  } |

## Tracker position

### General

This subclause specifies a sensor data type, which describes the position of the tracker.

### Syntax

|  |
| --- |
| "trackerPositionSensorData": {  "type": "object",  "properties": {  "position": {  "$ref": "#/definitions/float3DVectorType"  },  "orientationInQuaternion": {  "$ref": "#/definitions/float4DVectorType"  }  }  "additionalProperties": false  }, |

### Semantics

The semantics of the trackerPositionSensorData:

| Name | Definition |
| --- | --- |
| trackerPositionSensorData | Tool for describing sensor data concerning the tracker sensor’s position. |
| position | It describes the position of the tracker in a large space measured in meters. The position is described in the order of x, y, and z. |
| orientationInQuaternion | It describes the orientation of the tracker in a large space. The quaternion is described in the order of x, y, z and w. |

### Examples

In this example, the measured tracker position has x, y, and z values of 12 meters, 20 meters, and 1.8 meters, respectively. And the sensor is oriented towards the x-axis and rotated 30 degrees around the x-axis.

|  |
| --- |
| {  “sensedInfoBaseAttributes”: {},  “trackerPositionSensorData ”: {  “position”: [12, 20, 1.8],  "orientationInQuaternion": [1.0, 0, 0, 30]  }  } |

## Tracker velocity

### General

This subclause specifies a sensor data type, which describes the velocity of the tracker.

### Syntax

|  |
| --- |
| "trackerVelocitySensorData": {  "type": "object",  "properties": {  "velocity": {  "$ref": "#/definitions/float3DVectorType"  },  "deltaOrientationInQuaternion": {  "$ref": "#/definitions/float4DVectorType"  }  }  "additionalProperties": false  }, |

### Semantics

The semantics of the tackerVelocitySensorData:

| *Name* | *Definition* |
| --- | --- |
| trackerVelocitySensorData | Tool for describing sensor data concerning the tracker’s velocity. |
| velocity | It describes the velocity of the tracker in a large space measured in meters/second. The velocity is described in the order of x-direction, y-direction, and z-direction. |
| rotationSpeedInQuaternion | It describes the rotation speed (delta orientation) of the tracker in a large space. The quaternion is described in the order of x, y, z and w. |

### Examples

In this example, the sensor sensed that it was moving at 1.2 m/s in the x-axis and 1.5 m/s in the y-axis.

|  |
| --- |
| {  “sensedInfoBaseAttributes”: {},  “trackerVelocitySensorData”: {  “velocity”: [1.2, 1.5, 0]  }  } |

## Tracker acceleration sensor

### General

This subclause specifies a sensor data type, which describes the acceleration of the tracker.

### Syntax

|  |
| --- |
| "trackerAccelerationSensorData": {  "type": "object",  "properties": {  "acceleration": {  "$ref": "#/definitions/float3DVectorType"  },  "rotationalAccelerationInQuaternion": {  "$ref": "#/definitions/float4DVectorType"  },  "unitTimeForRA": {  "type": "number",  },  }  "additionalProperties": false  }, |

### Semantics

The semantics of the trackerAccelerationSensorData:

| *Name* | *Definition* |
| --- | --- |
| trackerAccelerationSensorData | Tool for describing sensor data concerning the tracker’s acceleration. |
| acceleration | It describes the acceleration of the tracker in a large space measured in meters/second/second. The acceleration is described in the order of x-direction, y-direction, and z-direction. |
| rotationalAccelerationInQuaternion | It describes the rotational acceleration (delta delta orientation) of the tracker in a large space. The quaternion is described in the order of x, y, z and w. |
| unitTimeForRA | It describes the unit time (in seconds) for calculating rotational acceleration. |

### Examples

In this example, the sensor sensed that it was accelerating at -0.5 m/s2 in the x-axis and 1.0 m/s2 in the z-axis.

|  |
| --- |
| {  “sensedInfoBaseAttributes”: {},  “trackerAccelerationSensorData”: {  “acceleration”: [-0.5, 0, 1]  }  } |