|  |  |
| --- | --- |
| Project | **Specification of Sensor Interface for Cyber and Physical World**<<https://sagroups.ieee.org/2888/> **>** |
| Title | **Input devices for a large space VR application** |
| DCN | **2888-21-0039-00-0001** |
| Date Submitted | **June 27th, 2021**  |
| Source(s) | Sang-Kyun Kim, goldmunt@gmail.com (Myongji University)Min Hyuk Jeong, jmh8900@gmail.com (Myongji University) |
| Re: |  |
| Abstract | This contribution illustrates the JSON schema structure for representing input device data in the physical world in a standardized data format. The semantics and examples of the input device data 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 input device data in a standardized data format in the physical world. The semantics and examples of the input device data are presented.

# Input device sensor data

## Button sensor

### General

This subclause specifies a sensor data type, which describes a button sensor.

### Syntax

|  |
| --- |
| "buttonSensorData": { "type": "object", "properties": { "messageTime": { "$ref": "#/definitions/messageTimeType" }, "button": { "type": "number" }, "state": { "type": "boolean" }  }, "additionalProperties": false }, |

### Semantics

The semantics of the buttonSensor:

|  |  |
| --- | --- |
| Name | Definition |
| buttonSensorData | Tool for describing button sensor data |
| messageTime | Describes sensor data acquisition time refer to messageTimeType |
| button | It describes the number of the button (numbered from zero). |
| state | Describes the current state of the button(True = on, False = off) |

### Examples

This example describes that the 7th button is on.

|  |
| --- |
| {“sensedInfoBaseAttributes”: {},“buttonSensorData”: { “button”: 7, “state”: true}} |

## Analog sensor

### General

This subclause specifies a sensor data type, which describes an analog sensor.

### Syntax

|  |
| --- |
| "analogSensorData": { "type": "object", "properties": { "messageTime": { "$ref": "#/definitions/messageTimeType" }, "numberOfChannels": { "type": "number" }, "channel": { "type": "array", "items": { "type": "number" } }  }, "additionalProperties": false }, |

### Semantics

The semantics of the analogSensor:

|  |  |
| --- | --- |
| Name | Definition |
| analogSensorData | Tool for describing the analog sensor data. |
| messageTime | It describes sensor information acquisition time refer to messageTimeType |
| numberOfChannels | It describes the number of channels. |
| channel | It describes analog data for each channel. It is repeated as many as the number of channels. |

### Examples

In this example, there are four channels. Analog data are 10, 12, 14, 16.

|  |
| --- |
| {“sensedInfoBaseAttributes”: {},“analogSensorData”: { “numberOfChannels”: 4, “channel”: [10, 12, 14, 16]}} |

## Dial sensor

### General

This subclause specifies a sensor data type, which describes a dial sensor.

### Syntax

|  |
| --- |
| "dialSensorData": { "type": "object", "properties": { "messageTime": { "$ref": "#/definitions/messageTimeType" }, "dial": { "type": "number" }, "change": { "type": "number" }  }, "additionalProperties": false }, |

### Semantics

The semantics of the dialSensor:

|  |  |
| --- | --- |
| Name | Definition |
| dialSensorData | Tool for describing the dial sensor data. |
| messageTime | It describes sensor information acquisition time refer to messageTimeType |
| dial | It describes the number of the dial sensor. |
| change | It describes the fraction of a revolution it changed in a clock-wise |

### Examples

In this example, the 4th dial rotated 13.3 degrees.

|  |
| --- |
| {“sensedInfoBaseAttributes”: {},“dialSensorData”: { “dial”: 3, “change”: 13.3}} |

## Haptic sensor

### General

This subclause specifies a sensor data type, which describes a haptic sensor.

### Syntax

|  |
| --- |
| "hapticSensorData": { "type": "object", "properties": { "messageTime": { "$ref": "#/definitions/messageTimeType" }, "force": { "$ref: "#/definitions/float3DVectorType" }, "position": { "$ref: "#/definitions/float3DVectorType" }, "quaternion": { "$ref: "#/definitions/float4DVectorType" } }, "additionalProperties": false }, |

### Semantics

The semantics of the hapticSensor:

|  |  |
| --- | --- |
| Name | Definition |
| dialSensorData | Tool for describing the haptic sensor data. |
| messageTime | It describes sensor information acquisition time refer to messageTimeType |
| force | It describes a sense of force in the three-axis direction from the force sensor (x, y, z). |
| position | It describes the position of the surface contact point of the virtual object (x, y, z). |
| quaternion | It describes a quaternion of the surface contact point of the virtual object (x, y, z, w). |

### Examples

In this example, the measured force from (30, 24, 10) is [10, 20, 30]

|  |
| --- |
| {“sensedInfoBaseAttributes”: {},“hapticSensorData”: { “force”: [10, 20, 30], "position": [30, 24, 10] }} |