OpenSpatial Specification

Version 0.6, 2015-01-19

This specification defines the OpenSpatial interface at the level of Bluetooth Smart services and their characteristics. Software libraries or developer kits for various platforms might present the information differently, for example by translating it to different data types, or by translating from relative to absolute coordinates, or by combining low-level information sequences into higher-level constructs.

Nod Openspatial Service

UUID: 0000febf-0000-1000-8000-00805f9b34fb

Nod Labs OpenSpatial Service is a Bluetooth Smart (Low Energy) service that transports motion and gesture data from a motion sensing device to a host device.

Multi-byte integer data types within characteristic values are in little-endian byte order. If the range shown is less than the containing type can hold, the meaning of out-of-range values is unspecified.

The service UUID can be expressed as the UUID16 0xfebf, as assigned by the Bluetooth SIG.

Characteristic: Position2D

UUID	Properties
00000206-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
X	int16	From -32768 to 32767 as described below
Y	int16	From -32768 to 32767 as described below

Position2D reports the projection of a sensor's motion onto a virtual 2-dimensional surface, similar in purpose to a "mouse pointer". X and Y are offsets from the position implied by the last Position2D report. The mapping function from motion to that surface, possibly including scale factors, coordi-

nate system origin, limits, and nonuniformities like acceleration, are unspecified but may be configurable by some means not specified herein.

Characteristic: Pose6D

UUID	Properties
00000205-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
X	int16	From -32768 to 32767 as described below
Y	int16	From -32768 to 32767 as described below
Z	int16	From -32768 to 32767 as described below
Roll	int16	From -25736 to 25736 representing -pi/2 to pi/2 radians
Pitch	int16	From -25736 to 25736 representing -pi/2 to pi/2 radians
Yaw	int16	From -25736 to 25736 representing -pi/2 to pi/2 radians

Pose6D reports translation and rotation of a sensor's motion.

X, Y, and Z are translational offsets from the position implied by the last Pose6D report, in a Cartesian coordinate space with unspecified origin and scaling. The axial orientation of that coordinate system is unspecified, but can generally be assumed to have the following attributes. Positive Z is up (away from the ground), positive X is away from the user, and positive Y is to the user's left. Alternatively, the X direction might be aligned with north, if that can be determined.

Roll, Pitch, and Yaw are absolute angles in a (1,2,3) Euler Angle space. The reference axes are such that Yaw is rotation about a vertical axis as defined by the Earth's gravity. The direction where yaw=0 is unspecified, but differences in yaw over time can nevertheless be used to determine rotational changes in that plane.

The X,Y,Z (translational) and Roll,Pitch,Yaw (rotational) coordinate systems are coupled in the sense that the X axis points in the Yaw=0 direction.

Certain sensor operations, such as major changes in operational status or specific user actions, might result in a change in the reference axes, for example by setting yaw to 0 and establishing a new direction for X.

Note - Euler Angle space is a bit-efficient means of conveying rotation information, but has well-known problems when used for computations. It is recommended that applications convert, using well-known formulas, the Euler Angle information to another representation, for example quaternions, prior to performing calculations. Transmitting rotations directly in quaternion format requires approximately twice as many bits for the same underlying angular resolution. The conversion from quaternions to Euler Angles and back can be done without loss of information. (Software libraries that provide access to this data may perform that conversion for the application.)

Characteristic: ButtonEvents

UUID	Properties
00000207-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
Buttons	uint16	See description

ButtonEvents reports changes in the activation state of on-off switches like touch surfaces and tactile switches.

Each switch's state changes are reported by a pair of adjacent bits within the uint16 value. The least-significant bit within that pair is 1 if that switch was activated since the last report, and the most-significant-bit within that pair is 1 if that switch was deactivated since the last report. If both bits are 1, it means that the switch was touched then immediately (since the last report) released.

The number of implemented switches and the assignment of specific physical switches to the numbered switches are product-dependent. In general, products should assign more-accessible or "primary" switches to lower switch numbers.

Switch	OnBit	OffBit
Switch0	0	1
Switch1	2	3
Switch2	4	5
Switch3	6	7
Switch4	8	9

Switch	OnBit	OffBit
Switch5	10	11
Switch6	12	13
Switch7	14	15

Characteristic: Gestures

UUID	Properties
00000208-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
direction	uint8	1=right 2=left 3=down 4=up 5=clockwise 6=counter-clockwise

Gestures reports the sensor's detection of predefined motion patterns.

Characteristic: Slides

UUID	Properties
0000020a-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
direction	uint8	1=left 2=right

Slides reports the sensor's detection of finger motion across a slider surface.

Characteristic: Motion6D

UUID	Properties
00000209-0000-1000-8000-a0e5e9000000	Read Notify

Value

Name	Type	Encoding
AccelX	int16	From -32767 to 32767 representing -N to N
AccelY	int16	From -32767 to 32767 representing -N to N
AccelZ	int16	From -32767 to 32767 representing -N to N
GyroX	int16	From -32767 to 32767 representing -M to M degrees/second
GyroY	int16	From -32767 to 32767 representing -M to M degrees/second
GyroZ	int16	From -32767 to 32767 representing -M to M degrees/second

Motion6D reports raw data from the motion sensor chip in its local coordinate system. The scale factors and clipping behavior for unrepresentable motions are product-dependent.