

# Parking sensor data feed specification

JULY 20, 2012

## Overview

SFpark uses parking sensor data for both real-time applications and historical analysis. This document summarizes the requirements for the parking sensor XML data feed.

Sensor vendors send parking data from a sensor management system to the SFpark data warehouse. These requirements are based on the current operating characteristics of the SFpark system, and could be modified to accommodate additional functionality and/or features in the future.

## Key elements

The following are the important elements of the parking sensor data feed. A full schema definition is in Appendix A:

- **Parking space ID (PSID).** The PSID is SFpark's unique database identifier for parking spaces. This identifies a spatial location that corresponds to a parking space. SFpark processes all data based on the PSID; correct associations are crucial.
- **Sensor ID.** SFMTA uses the Post ID in normal operations to identify parking spaces. Post ID's are visible on the street – every single space meter has a decal with the Post ID, and Post IDs can be derived for multi-space meters. The format for all PostID's is SSS-BBNNX where:
  - SSS is 3 digit street ID
  - BB is 2 digit block #
  - NN is the closest approx address # on the block. (Odd vs. even NN value indicates blockface.)
  - X for POST\_ID's that are assigned to individual parking spaces are generally always 0. It may be used for individual spaces of multi-space meters so the first 8 digits correspond to the MS\_PAY\_STATION\_ID. There may be other circumstances, even for SS meters where BBX are all used together for sequential numbering assignments.
- **Event type.** This is the parking space status being communicated.
- **Event time.** This is the time that the parking event occurred. Generally, timestamps for SS events are when the car stops moving within the space, and SE events are when vehicles clear the space.
- **Transmission time.** This is the time that the vendor has sent the event to the SFpark data warehouse. SFpark will populate its database with the time the event was received, and use this to evaluate latency.

## Event types

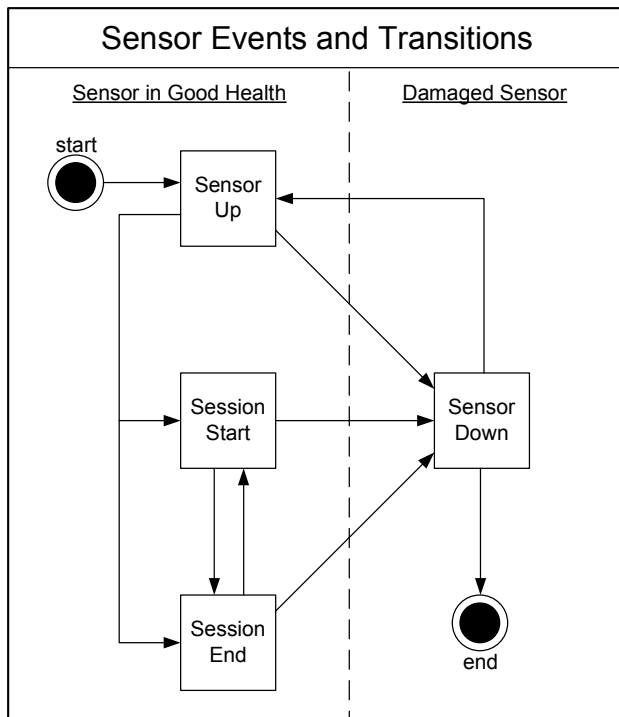
There are five types of events: two communicate parking status and three communicate operational status:

- **Session Start (SS).** A vehicle has arrived in the parking space and has started a new parking session.
- **Session End (SE).** A vehicle has departed from the parking space and has completed a parking session.
- **Sensor Down (SD).** The sensing and communication ability of the sensor or sensors in a space have been determined to be faulty.
- **Sensor Up (SU).** The sensor has been brought online and is ready for service.
- **Heartbeat (HB).** This message is a positive affirmation of a sensor's operational status.

All Sensor Event messages are sent by the Sensor Management Service in real time as the events are detected.

## Logical rules

The following diagram illustrates the allowed flow of events:



- A parking session will always begin with a SS event and conclude with a SE event (and have a matching session ID). Thus, the sequence of events will be **SS -> SE -> SS**. A SE event may occur as close as one second to either the preceding or following SS events.
- The exception to this rule occurs in the event of a sensor failure. In this scenario a SD event message will be sent by the Sensor Management Service.
- A SD event will be sent whenever a sensor enters a period of irrevocable outage. A SD event may occur at any time regardless of the previous state. The sensor is to remain in this state until it is declared healthy by the Sensor Management Service after being repaired or replaced.
- No events are to be sent with timestamps that are equal to or between the SD and SU events.
- When a sensor is again providing accurate data for the parking space a SU event will be sent.
- A sensor in the SD state must always transition to SU before it resumes sending parking session events.
- No synthetic events are to be sent as a result of an outage, and no events are to be suppressed after an outage. As such the following sequences of event are considered to be valid:

```
SS -> SD -> SU -> SS
SS -> SD -> SU -> SE
SE -> SD -> SU -> SE
SE -> SD -> SU -> SS
```

- Sensor Down and Sensor Up event are not to be sent to indicate data delay or communications issues between the sensor(s) and the Sensor Management Service.
- No two events shall have the same sensor ID and event time. An event will be rejected if it matches the event time and sensor ID of a previously received event and the event types conflict (e.g., a SS and SE with the same event time and same sensor ID). In other words, the event with the earliest transmission time will be considered valid and all subsequent simultaneous conflicting events will be rejected.

## Towaway filter

SFpark provides the sensor vendor with a list of Post IDs and corresponding times where peak-hour tow-away regulations are enforced. No events are sent during the towaway period.

## Appendix A: Schema description

Below is the schema description along with key element types. Some optional elements and type definitions have been removed for brevity.

```

<xs:schema targetNamespace="http://www.sfmta.com/xsd/parking"
  xmlns="http://www.sfmta.com/xsd/parking" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="SENSOR" type="SENSOR"/>

  <xs:complexType name="SENSOR">
    <xs:all>
      <xs:element name="VENDOR_ID" type="PosIntegerType" minOccurs="1"
maxOccurs="1"/>
      <xs:element name="TRANSMISSION_ID" type="PosLongType" minOccurs="1"
maxOccurs="1"/>
      <xs:element name="TRANSMISSION_DATETIME" type="SFParkDateFormat"
minOccurs="1" maxOccurs="1"/>
      <xs:element name="EVENT_TYPE" type="EventType" minOccurs="1"
maxOccurs="1"/>
      <xs:element name="EVENT_TIME" type="SFParkDateFormat" minOccurs="1"
maxOccurs="1"/>
      <xs:element name="SENSOR_TYPE" type="SensorType" minOccurs="1"
maxOccurs="1" />

      <xs:element name="METERED_SPACE" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:all>
            <xs:element name="PS_ID" type="PosIntegerType"
minOccurs="0" maxOccurs="1" />
            <xs:element name="SENSOR_ID" type="SensorIdType"
minOccurs="1" maxOccurs="1" />
            <xs:element name="SESSION_ID" type="xs:long"
minOccurs="0" maxOccurs="1"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:all>
  </xs:complexType>

  <xs:simpleType name="EventType">

```

```

<xs:restriction base="xs:string">
    <xs:enumeration value="SS"/> <!-- Session Start -->
    <xs:enumeration value="SE"/> <!-- Session End -->
    <xs:enumeration value="SD"/> <!-- Sensor Down -->
    <xs:enumeration value="SU"/> <!-- Sensor Up -->
    <xs:enumeration value="HB"/> <!-- Heart beat -->
</xs:restriction>
</xs:simpleType>
</xs:schema>

```

**Descriptions for Child Elements of the <SENSOR> Element**

Element Name	Data Type	Description
VENDOR_ID	PosIntegerType	Assigned by SFpark
TRANSMISSION_ID	PosLongType	Unique Positive number - Generated by Proposer Transmission ID will utilize underlying sensor event unique ID followed by a suffix digit representing the type and/or state of the message as follows: 0 – Original Transmission 1 – Retransmitted SS or SE 2 – SU, SD, or HB
TRANSMISSION_DATETIME	DateTime	Generated by Proposer Date and time of transmission of the message Time in UTC time zone Standard Oracle format to second "YYYY-MM-DD HH:MM:SS" e.g. "2010-02-11 14:32:22"
EVENT_TYPE	Enumeration <string>	Generated by Proposer <b>SS:</b> Session Start A car pulls into a metered parking space <b>SE:</b> Session End A car pulls out of a metered parking space <b>SD:</b> Sensor Down A Sensor becomes inactive <b>SU:</b> Sensor Up A Sensor becomes active <b>HB:</b> Heartbeat Heartbeat message indicating the sensor is operational

METERED_SPACE		Each SENSOR element may contain 1 and only 1 of the METERED_SPACE elements.
PS_ID	PosIntegerType	Parking Space ID - Assigned by SFpark. Shall be the unique identifier for the metered parking space <b>99999</b>
SENSOR_ID	String	Assigned by SFpark. Shall be the Post ID for the metered parking space. <b>999-99999</b>
SESSION_ID	PosLongType	Positive long number. Generated by Proposer. It should tie a particular SS event to the corresponding SE event.
EVENT_TIME	DateTime	Time in UTC time zone Date and time of the sensor event Standard Oracle format to second "YYYY-MM-DD HH:MM:SS" e.g. "2010-02-11 14:32:22" No two events for a sensor, with the exception of the HB, should have the same event time.

**Example of a <SENSOR> Element**

```

<SENSOR>
  <VENDOR_ID>1</ VENDOR_ID>
  <TRANSMISSION_ID>99999</TRANSMISSION_ID>
  <TRANSMISSION_DATETIME>2010-12-31 13:00:00</TRANSMISSION_DATETIME>
  <EVENT_TYPE>SS</EVENT_TYPE>
  <METERED_SPACE>
    <PS_ID>99999</PS_ID>
    <SENSOR_ID>999-99999</SENSOR_ID>
    <SESSION_ID>99999</SESSION_ID>
  </METERED_SPACE>
  <EVENT_TIME>2010-12-31 12:58:00</EVENT_TIME>
</SENSOR >

```