



# Technical Reference Document

Version 1.1

This document briefly explains the APIs exposed by GPSit's tracking system. It includes new functions recently implemented to ease locating and tracking of a GPSit tracking device provided by Sendum.

GPSit supports following devices through its API:

- Sendum VT100
- Sendum PT200
- Sendum PT300
- Sendum GT200
- Sendum GT300 (Support coming soon)
- GPSit's own SmartBoard

Each device has a different set of commands and API functions unify common commands into functions. Device specific commands have their own API functions through which these commands can be configured.

## SOAP 1.2 Interfaces

### Most commonly used SOAP service

WSDL	<a href="http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?WSDL">http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?WSDL</a>
Interface	<a href="http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx">http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx</a>

This web service supports all the common operations needed for basic tracking. Each device needs to be in GPSit database to be able to utilize the web services. Most common operations include the following:

- Locating a device.
- Tracking a device in real time.
- Getting battery and network status from the device.

Most of the aforementioned operations require sending a command to the device and followed by polling the service to check if there was a response from the device or not e.g. locating a device requires two API calls; First call to send a message to the device followed by a second call to fetch the fresh location that the device just sent it in.

## Locating a device

This operation is performed by first calling the following API function

<http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?op=LocateNow>

This function accepts device MDN and then fetches device configuration from the database and according to the BREW firmware, it generates a command which is then sent using either an SMTP server or Carrier's SMS web services. Once the device receives the message, it fetches its GPS coordinates using satellites/cell towers and sends either a successful location or a location error back to GPSit's server.

Upon receiving the response from the device, parsing and storing is done by the TCP/IP listener. To fetch this successful location, a second API call is made which is as following

<http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?op=GetLastLocateNowTrackDataPoint>

This call should be made either by polling at every few seconds interval or roughly calling it just once roughly after 45-60 seconds from the first request. Response from this function should look like this

```
<?xml version="1.0" encoding="utf-8" ?>
- <TrackDataStruct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" :
  <trackUID>1296470</trackUID>
  <devicePID>6784787239</devicePID>
  <deviceName>New York-57</deviceName>
  <latitude>32.7187881469727</latitude>
  <longitude>-117.151206970215</longitude>
  <accuracy>64</accuracy>
  <date>8/27/2012</date>
  <time>11:04:21 AM</time>
  <desc>8429216</desc>
  <locateType>MSAL</locateType>
  <reported>>true</reported>
  <timestamp>1346090627</timestamp>
  <error>0</error>
</TrackDataStruct>
```

Note that error is 0. If it is any integer other than 0, that means there was an error trying to fetch location of the device. Possible values are 0, 1, 2 and 4.

## Tracking a device in real time

This operation also requires two API calls. Following is the first API function that needs to be called

<http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?op=LocateRealTime>

This function accepts device MDN, interval and duration as its parameters.

This function sends a REPORT,UPL,<interval> message to the device. <interval> is how often the device should report its location to the server. <duration> is used by the software behind the scenes. <duration> is for how long should the tracking continue.

During this tracking, the device reports every <interval> seconds for <duration> seconds. When duration ends, software sends out a UPL,STOP message to stop the tracking.

To fetch response messages sent by the device, a recurring second call needs to be made by polling at least every interval/2 seconds till the end of tracking.

<http://trackfusion.gpsit.com/FindandSee2/bllWebServices/webServices.asmx?op=LastTrackDataPoint>

Sample response looks like the one below

```
<?xml version="1.0" encoding="utf-8" ?>
- <TrackDataStruct xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  <trackUID>1293084</trackUID>
  <devicePID>6784787239</devicePID>
  <deviceName>New York-57</deviceName>
  <latitude>32.7182350158691</latitude>
  <longitude>-117.150756835938</longitude>
  <accuracy>192</accuracy>
  <date>8/13/2012</date>
  <time>4:26:39 PM</time>
  <desc>8364447</desc>
  <locateType>MSAL</locateType>
  <reported>true</reported>
  <timestamp>1344900600</timestamp>
  <error>0</error>
</TrackDataStruct>
```

Error field indicates a location or network error. Possible values are 0, 1, 2, and 4.

## Geofence service

WSDL	<a href="http://trackfusion.gpsit.com/FindandSee2/bllWebServices/GeofenceService.asmx?WSDL">http://trackfusion.gpsit.com/FindandSee2/bllWebServices/GeofenceService.asmx?WSDL</a>
Interface	<a href="http://trackfusion.gpsit.com/FindandSee2/bllWebServices/GeofenceService.asmx">http://trackfusion.gpsit.com/FindandSee2/bllWebServices/GeofenceService.asmx</a>

This web service is used to create and manage geofences. Geofences are virtual fences around a center geo location with a set radius. Device tracks at a set frequency and the software checks if there was a breach or not. If there are configurations set, upon breach, the user will be alerted as well device can be set to track at a faster interval. Breach can be upon entry or upon exit.

Multiple geofences at different locations can be created for a single device.

### Creating a Geofence

To create a geofence, following service needs to be called

<http://trackfusion.gpsit.com/FindandSee2/bllWebServices/GeofenceService.asmx?op=AddGeofence>

It requires following parameters

Parameter	Value
MDN:	<input type="text"/>
FenceName:	<input type="text"/>
FenceRadius:	<input type="text"/>
FenceLat:	<input type="text"/>
FenceLon:	<input type="text"/>
StartDate:	<input type="text"/>
EndDate:	<input type="text"/>
Interval:	<input type="text"/>
Color:	<input type="text"/>
FenceGroup:	<input type="text"/>
AlertType:	<input type="text"/>
TrkOn:	<input type="text"/>
TrkOff:	<input type="text"/>
TrkFreq:	<input type="text"/>
ZoneCond:	<input type="text"/>

<Interval> is how often the device should report its location to be able to find out if there was a geofence breach or not.

<FenceGroup> parameter can safely be ignored.

<AlertType> defines how to alert the user i.e. upon exit, upon entry or upon both entry and exit. Possible values are 0, 1, 2 and 3.

<TrkOn> defines when to start tracking at different frequency e.g. track at a faster frequency if the breach is detected. Possible values are ONENTRY or ONEXIT.

<TrkOff> defines when to stop tracking which was started by TrkOn parameter. Possible values include ONENTRY, ONEXIT.

<TrkFreq> indicates the frequency at which to track the device when TrkOn is set.

<ZoneCond> parameter can safely be ignored.