Tizen Vehicle Information
Web API Specification

Kevron Rees
April 16, 2013
Contents

- Overview
- Use-Cases
- Examples
- Data Types
- Events
- Future plans
Overview

- Purpose:
  - to enable Tizen IVI developers with rich access to vehicle information
Use cases

1. Developer wants to write a tachometer application in html5
2. Developer wants to write an application that controls the HVAC system
3. Mechanic wants to retrieve the vehicle information from last week
1. Developer wants to write a tachometer application in html5

- get ()
Example

1. Developer wants to write a tachometer application in html5

```javascript
navigator.vehicle.get("VehicleSpeed", onsuccess, onerror);

function onsuccess(value) {
  window.console.log(value.VehicleSpeed);
}

function onerror(e) {
  window.console.error(e.message);
}
```
2. Developer wants to write an application that controls the HVAC system
   - set ()

```cpp
/**
 * \brief set the given property to value
 * \arg DOMString property property to set
 * \arg VehiclePropertyType value value to set
 * \arg VehiclePropertyCallback successCallback callback if operation is successful
 * \arg VehiclePropertyErrorCallback errorCallback callback if error has been called.
 */
set(DOMString property, VehiclePropertyType value, optional VehiclePropertyCallback successCallback, optional VehiclePropertyErrorCallback errorCallback);
```
Example

2. Developer wants to write an application that controls the HVAC system

```javascript
navigator.vehicle.get("HVAC", onsuccess, onerror);

function onsuccess(value) {
  var hvacsettings = value;

  // send air out the front vents and defroster
  value.AirflowDirection = value.AIRFLOWDIRECTION_FRONT | value.AIRFLOWDIRECTION_DEFROSTER;

  navigator.vehicle.set("HVAC", value, onsetsuccess, onerror);
}

function onerror(e) {
  window.console.error(e.message);
}

function onsetsuccess() {
  window.console.log("success!");
}
```
3. Mechanic wants to retrieve the vehicle information from last week.
- getHistory()
Example

3. Mechanic wants to retrieve the vehicle information from last week.

```javascript
var startDate = new Date("April 5, 2013 11:13:00");
var endDate = new Date("April 10, 2013 11:13:00");

navigator.vehicle.getHistory("VehicleSpeed", startDate, endDate, onsuccess)

function onsuccess(values) {
  window.console.log(values.count())
}
```
Data Types

```c
interface Acceleration : VehiclePropertyType {

    /** X
     * \brief Must return acceleration on the "X" axis as 1/1000 G (gravitational force)
     **/
    readonly attribute unsigned long X;

    /** Y
     * \brief Must return acceleration on the "Y" axis as 1/1000 G (gravitational force)
     **/
    readonly attribute unsigned long Y;

    /** Z
     * \brief Must return acceleration on the "Z" axis as 1/1000 G (gravitational force)
     **/
    readonly attribute unsigned long Z;
};
```
Events

```javascript
navigator.vehicle.addEventListener("VehicleSpeed", vehicleSpeedHandler, null);

function vehicleSpeedHandler(data) {
    window.console.log(data.VehicleSpeed + "kph")
}
```
Future

- Transfer from WAC-style callbacks to W3C style
  - Use DOMFuture

```javascript
interface Vehicle {
  ...
  DOMFuture speed; // async
};

navigator.vehicle.speed.then(onSuccess, onerror);

function onSuccess(value) {
  window.console.log(value.VehicleSpeed);
}

function onerror(e) {
  window.console.error(e.message);
}
```
Resources

Tizen Vehicle API draft specification:


Draft WebIDL:

https://raw.githubusercontent.com/otcshare/automotive-message-broker/master/docs/amb.idl

Contact:

Kevron Rees: kevron.m.rees@intel.com