

# Live E! Profile Registration I/F Specification

--- ProfileManagement200703 ---

Hideya Ochiai, The University of Tokyo

Satoshi Matsuura, NAIST

Created: 2008-10-10

Last Updated: 2009-02-28

## 1. Abstract

This specification defines an interface(I/F) for managing sensor profiles in Live E! data platform. Live E! service has (1)profile registration I/F, (2)data upload I/F and (3) data retrieval I/F (Figure 1). This document describes only (1) profile registration I/F.

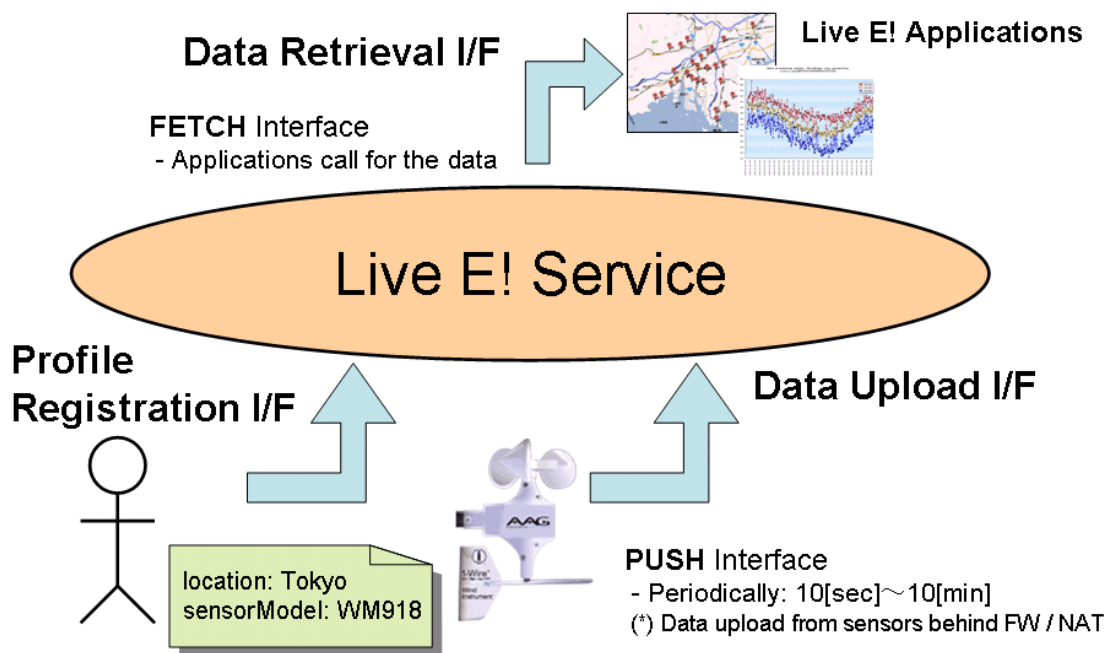


Fig. 1: Live E! Service Overview

Live E! service is implemented by a collection of servers, which are networked among them in overlay networking manner. These three types of I/Fs are attached to every server node, via which sensors and users upload/download sensor readings.

## 2. Profile Registration I/F

Profile Registration I/F is deployed as an SOAP/XML Web service. We define the name of service and methods as follows.

### <<Service Name>>

ProfileManagement200703

### <<Methods>>

String getAvailableLocales();

String getProfile(String id);

String getProfileAll(String timezone,String locale);

String getProfileSchema();

String setProfile(String profile);

String updatePassword(String id,String old\_passwd, String new\_passwd);

### How to call the service: (an example in perl script)

```
#!/usr/bin/perl
use strict;
use warnings;
use SOAP::Lite;

# formatting of the sensor profile
$profile= .... ;

# data transmission
my $server = SOAP::Lite -> service('http://live-e2.hongo.wide.ad.jp/axis/services/ProfileManagement200703?wsdl');
my $res = $server-> setProfile($profile);

# Response checking with error management
...
```

## 2.1. String `getAvailableLocales()`;

`getAvailableLocales` returns system defined language locale codes.

### <<Return Value>>

`getAvailableLocales` returns the following values.

If succeeded,

- Available language locales

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

`ERROR_TYPE` is one of the following values.

- internal: Unknown Error

<< An example of the return value >>

```
<?xml version="1.0" encoding="UTF-8"?>
<locales xmlns="http://live-e.org/Schema/2007/03/">
  <locale code="chi" name="Chinese"/>
  <locale code="eng" name="English"/>
  <locale code="fre" name="French"/>
  <locale code="ger" name="German"/>
  <locale code="jpn" name="Japanese"/>
  <locale code="kor" name="Korean"/>
  <locale code="tha" name="Thai"/>
</locales>
```

## 2.2. String getProfile(String id);

getProfile returns the profile of the specified combined sensor ID.

### <<Return Value>>

getProfile returns the following values

If succeeded,

- The profile of the specified sensor.

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

ERROR\_TYPE is one of the following values.

- sensorNotFound: The specified sensor do not exist.
- internal: Unknown error

<< An example of the return value >>

The result of getProfile("soum.co.jp/WM918/mito01");

```
<?xml version="1.0" encoding="UTF-8"?>
<sensorGroup address_eng="1-6-8 Goken-cho, Mito-shi, Ibaraki 310-0063 JAPAN" address_jpn=""
class="combined" id="soum.co.jp/WM918/mito01/" latitude="36.38055" location_eng="ART TOWER MITO"
location_jpn="" longitude="140.46585" sensorModel="WM918" sensorVendor="AmbientWeather"
xmlns="http://live-e.org/DataType/2007/03/">
<sensor id="soum.co.jp/WM918/mito01/Temperature" sensorType="Temperature"/>
<sensor id="soum.co.jp/WM918/mito01/Humidity" sensorType="Humidity"/>
<sensor id="soum.co.jp/WM918/mito01/Pressure" sensorType="Pressure"/>
<sensor id="soum.co.jp/WM918/mito01/RainFall" sensorType="RainFall"/>
<sensor id="soum.co.jp/WM918/mito01/WindDir" sensorType="WindDir"/>
<sensor id="soum.co.jp/WM918/mito01/WindSpeed" sensorType="WindSpeed"/>
</sensorGroup>
```

### 2.3. String `getProfileAll(String timezone, String locale)`;

`getProfileAll` returns all the sensor profiles registered in the server with specified timezone and locale. This method should be called by operational users for just listing up sensor list. For the detailed profile retrieval, see Section 2.2 String `getProfile(String id)`.

#### <<Return Value>>

`getProfileAll` returns the following value.

If succeeded,

- All the profiles registered in the server.

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

`ERROR_TYPE` is one of the following values.

- `outOfSize`: Too much sensor collections (= exceeded `maxSensorCollection`)
- `internal`: Unknown error

## 2.4. String getProfileSchema();

getProfileSchema returns the schema of profiles defined in the global Live E! system. Every sensor profile should be verified before registering into the server by this profile schema.

### <<Return Value>>

getProfileSchema returns the following values.

If succeeded,

- The current profile schema

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

ERROR\_TYPE is,

- internal: Unknown error

### <<An example of the return value>>

```
<?xml version="1.0" encoding="UTF-8"?>
<profileSchema xmlns="http://live-e.org/Schema/2007/03/">
  <schema class="combined|element" name="longitude" type="float" value=".*"/>
  <schema class="combined|element" name="sensorVendor" type="string"
    value="Vaisala|AmbientWeather|AAG|Davis|Ubiteq|MatsushitaDenko|TriState"/>
  <schema class="value" name="time" type="time" value=".*"/>
  <schema class="element" name="accuracy" type="float" value=".*"/>
  <schema class="combined" name="road" type="string" value=".*"/>
  <schema class="combined" name="cameraUrl" type="string" value=".*"/>
  <schema class="combined|element" name="altitude" type="float" value=".*"/>
  <schema class="element" name="sensorType" type="string"
    value="Temperature|Humidity|Pressure|DayRainFall
    |RainFall|WindSpeed|WindDir|CO2|SolarPower|Illuminance
    |AccelerationX|AccelerationY|Soil_Moisture|Soil_Temperature|Solar_Radiation"/>
  <schema class="element" name="error" type="float" value=".*"/>
  <schema class="combined|element" name="sensorModel" type="string"
    value="WXT510|WM918|WM928|WMR968|VantagePRO|WSN-100X|FS-Va-01|PICNICv12|TA18515"/>
  <schema class="combined|element" name="latitude" type="float" value=".*"/>
  <schema class="combined" multilanguage="true" name="address" type="string"
    value=".*"/>
  <schema class="combined" multilanguage="true" name="location" type="string"
    value=".*"/>
  <schema class="combined|element" name="gAltitude" type="float" value=".*"/>
</profileSchema>
```

## 2.5. String setProfile(String profile);

setProfile registers the specified sensor profile into the server. The access password should be encoded into BASE64 and included into the profile form with authorization attribute.

### The format of profile

```
<?xml version=" 1.0" encoding=" UTF-8" ?>
<sensorGroup class=" combined" id=" SENSOR_COMBINED_ID" authorization ="
BASE64_ENCODED_PASSWD" [(COMBINED_ATTR)*] xmlns=" http://live-e.org/DataType/2007/03/" >
  <sensor id=" SENSOR_ELEMENT_ID" [(ELEMENT_ATTR)*] />
  ... Repeat of <sensor /> ...
</sensorGroup>
```

### The semantics of the data field

Data Field	Semantics
SENSOR_COMBINED_ID	Identifier (ID) of the combined sensor
SENSOR_ELEMENT_ID	Identifier (ID) of the element sensor
BASE64_ENCODED_PASSWD	Access password for the ID. It should be encoded by base64.
COMBINED_ATTR	An attribute, a "name=value" pair, which schema is defined by profile schema for combined sensors. E.g., (1) The location of the value observed: latitude="35.314" longitude="135.231" E.g., (2) The status of the value observed: error="overflow"
ELEMENT_ATTR	An attribute, a "name=value" pair, which schema is defined by profile schema for element sensors. E.g., (1) The type of the element sensor, sensorType="Temperature" E.g., (2) The available observation range, valueMax="40.0" valueMin="-10.0" E.g., (3) The measurement type measurementType="Celsius"

- Multi-Language Support Attributes

We can set attribute values in each language for the attributes which are `multilanguage="true"` in the profile schema. The attribute name that should be written in XML should be formatted as follows: `ATTR=NAME + LANG_CODE`.

For example,

When the attribute "address" is defined in the profile schema as follows,

```
<schema class="combined" name="address" multilanguage="true" type="string" value=".*" />
```

The "address" is an attribute for combined sensors which should be handled as a character-string and has multiple-language expressions. The `COMBINED_ATTR` formatted in the data field must be written using `jpn` (for Japanese) and `eng`(for English),

```
address_jpn="東京都文京区本郷 7-3-1 東京大学 工 2 号館"
```

```
address_eng="Bld. 2 of Engineering Dpt. University of Tokyo, 7-3-1 Hongo, Bunkyo, Tokyo, Japan"
```

The language codes are provided by `getAvailableLocales()`;

## <<Return Value>>

`setProfile` returns the following values.

If succeeded,

```
<true />
```

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

`ERROR_TYPE` is one of the following values.

- authentication: Authentication error
- internal: Unknown error

If any error has occurred, the requested profile will not be registered.

## 2.6. String updatePassword(String id, String old\_passwd, String new\_passwd);

updatePassword updates the registered access password by combined sensor ID.

The parameters and the meanings

- String id – The target combined sensor ID.
- String old\_passwd – BASE64 encoded old password for the id.
- String new\_passwd – BASE64 encoded new password for the id.

### <<Return Value>>

updatePassword returns the following values.

If succeeded,

```
<true />
```

If any error has occurred,

```
<error type=" ERROR_TYPE" xmlns=" http:// live-e.org/Error/2007/03/" > ERROR_MESSAGE</error>
```

ERROR\_TYPE is one of the following values.

- authentication: Authentication error
- internal: Unknown error

If any error has occurred, the password will not be updated.