

Tutorial: Live E! Server Operation

Aug 22st 2007

Live E! Technical Working Group

Outline

This tutorial describes the outline and operational manual of wide-area sensor network (Live E!). Live E! system is composed of autonomously distributed servers and sensors which are administrated by multiple organizations. Sensor data sampled by these organizations can be shared in the community for allowing new applications: e.g., ubiquitous use, facility management, agriculture, natural disaster and education.

Section 1: Live E! System Overview

Live E! system is composed of distributed servers which autonomously form a community and share sensor data over the Internet. These distributed servers are administrated by multiple organizations that are interested in and actually operate sensors. Sensor data are shared in the Live E! server network and any user who joins in the community can retrieve sensor data from the network system.

Section 2 describes how to install a Live E! server: i.e., (1)download Live E! server package, (2)install the package and (3)join in the Live E! server network.

Section 3 gives a brief introduction to server operation. Making accounts for authorizing sensors is required, and this section shows how to make an account, authenticate and authorize sensors. After accounting sensors, profiles and data can be registered and uploaded. Sensor profile gives additional information to sensors, which describes location of the sensor, type of the sensor and other information. Profile is used for not only giving additional information but also indexing sensor from various application aspects in the global server network.

Section 4 shows how to retrieve sensor data from the global Live E! server network briefly. Every server can retrieve data from the global network; i.e., the target servers that contain data which matches user query can be automatically selected and data can be retrieved from each server and merged. Data retrieval interface is provided by SOAP web service. Sensor application developers can easily create application software using the web interface.

Section 5 gives a server redundancy technique to get robustness and service availability of an operational unit. The same data and service can be stored and provided by redundant servers. These servers are synchronized by master / slave architecture.

Section 6 describes schema-based profile management for improving profile consistency and application layer interoperability. Application objects which are described on sensor profiles are administrated by a global schema.

Section 7 gives instructions that any organization should obey when developing new sensors for Live E!. The new sensor must be approved and registered at Live E! authority to use it in the global community. The new sensor will be included in the profile schema after the approval.

Section 2: Server Installation

The following is the installation steps of a Live E! server.

- (1) Base system installation
- (2) Live E! system installation
- (3) Live E! server configuration

2.1. Base system installation

The installation of base software components on Ubuntu Linux:

- JDK 5: Java Developers Kit
- Apache 2: Web Server
- Tomcat 5.5: Web Servlet Engine
- Axis 1.4: SOAP Web Service Engine
- PostgreSQL 8.1: Database Management System

2.1.1. JDK 5 Installation

```
$ sudo apt-get install sun-java5-jdk
```

JAVA_HOME environment setting:

```
$ sudo vi /etc/profile  
JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun/  
export JAVA_HOME
```

Confirmation:

```
$ source /etc/profile
```

```
$ echo $JAVA_HOME
```

2.1.2. Apache2 Installation

```
$ sudo apt-get install apache2
```

2.1.3. Tomcat5.5 Installation

```
$ sudo apt-get install tomcat5.5 tomcat5.5-admin tomcat5.5-webapps
```

```
$ cd /var/lib/tomcat5.5/
```

```
$ sudo mv catalina.out catalina.out.old
```

```
$ sudo touch catalina.out
```

```
$ sudo chown tomcat5.nogroup catalina.out
```

```
$ sudo /etc/init.d/tomcat5.5 restart
```

Note: After the installation of tomcat5.5, tomcat5.5-admin and tomcat5.5-webapps, change permission of catalina.out, and reboot tomcat5.5.

Next, delete files as follows:

```
$ sudo rm /usr/share/tomcat5.5/common/endorsed/xercesImpl.jar
```

```
$ sudo rm /usr/share/tomcat5.5/common/endorsed/xml-apis.jar
```

These files cause failure in instantiating XPathFactory java class.

Confirm that the tomcat service is available by accessing “<http://localhost:8080/>” or “<http://localhost:8180/>” with your web browser.

Disable tomcat security mode for Axis web service engine:

```
$ sudo vi /etc/default/tomcat5.5
```

Replace the following sentence,

```
#TOMCAT5_SECURITY=yes
```

by

```
TOMCAT5_SECURITY=no
```

2.1.4. Axis1.4 Installation

Download and extract axis-bin-1_4.tar.gz from “<http://ws.apache.org/axis/>”.

```
$ cd
```

```
$ wget http://www.apache.org/dist/ws/axis/1_4/axis-bin-1_4.tar.gz
$ cd /usr/local/
$ sudo tar zvxf ----/axis-bin-1_4.tar.gz # The downloaded axis package
$ sudo ln -s /usr/local/axis-bin-1_4 /usr/local/axis
```

Make a directory copy of /usr/local/axis/webapps/axis to /var/lib/tomcat/webapps/, and change its permission setting.

```
$ sudo cp -r /usr/local/axis/webapps/axis /var/lib/tomcat/webapps/
$ sudo chown -R tomcat5.nogroup /var/lib/tomcat/webapps/axis
```

Reboot tomcat:

```
$ sudo /etc/init.d/tomcat5.5 restart
```

Confirmation of Axis execution (just the confirmation of execution, the confirmation of complete installation is still required after installation of jar files):

Refer to “<http://localhost:8080/axis/>” or “<http://localhost:8180/axis/>” with your web browser.

Installation of required and optional JAR files.

. JavaBeans Activation Framework 1.1 (activation.jar)

Download jaf-1_1-fr.zip from:

<http://java.sun.com/products/javabeans/jaf/downloads/index.html>

Copy activation.jar contained in the ZIP file to:

`/usr/share/tomcat5.5/common/lib/`

```
$ sudo cp activation.jar /usr/share/tomcat5.5/common/lib/.
```

. JavaMail API 1.4 (mail.jar)

Download javamail-1_4.zip from:

<http://java.sun.com/products/javamail/downloads/index.html>

Copy mail.jar contained in the ZIP file to:

`/usr/share/tomcat5.5/common/lib/`

```
$ sudo cp mail.jar /usr/share/tomcat5.5/common/lib/.
```

. XML Security (xmlsec.jar)

Download xml-security-bin-1_4_1.zip from:

<http://xml.apache.org/security/dist/java-library/>
(<http://xml.apache.org/security/download.html>)

Copy xmlsec-1.4.1.jar contained in the ZIP file to

`/usr/share/tomcat5.5/common/lib/`

```
$ sudo cp xmlsec-x.x.x.jar /usr/share/tomcat5.5/common/lib/.
```

Restart tomcat:

Check axis installation status by its validation page, which is linked from “<http://localhost:8080/axis/>” or “<http://localhost:8180/axis/>”.

2.1.5. Apache tomcat connector (mod_jk2) installation

```
$ sudo apt-get install libapache2-mod-jk
```

```
$ sudo vi /etc/apache2/mods-enabled/jk.conf
```

Describe the following configuration script into `jk.conf`:

```
JkLogFile      /var/log/apache2/mod_jk.log
JkLogLevel     error
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"

JkMount /axis/* ajp13w
JkWorkerProperty worker.list=ajp13w
JkWorkerProperty worker.ajp13w.type=ajp13
JkWorkerProperty worker.ajp13w.host=localhost
JkWorkerProperty worker.ajp13w.port=8009
```

Restart Apache2:

```
$ sudo /etc/init.d/apache2 restart
```

Confirm the connection from the apache to the axis by checking “<http://localhost/axis/>” with your web browser.

2.1.6. PostgreSQL 8.1 Installation

```
$ sudo apt-get install postgresql-8.1
```

Configuration of PostgreSQL:

Modify some sentences at the end of `/etc/postgresql/8.1/main/pg_hba.conf` as follows:
i.e., replace “indentsameuser” and “md5” with “trust”.

local	all	postgres		trust	
#	TYPE	DATABASE	USER	CIDR-ADDRESS	METHOD
local	all	all		trust	
#	IPv4 local connections				
host	all	all	127.0.0.1/32	trust	
#	IPv6 local connections				
host	all	all	::1/128	trust	

Restart PostgreSQL:

```
$ sudo /etc/init.d/postgresql-8.1 restart
```

Installation of PostgreSQL JDBC Driver:

Download `postgresql-8.1-409.jdbc3.jar` from <http://jdbc.postgresql.org/download.html> and copy it to `/usr/share/tomcat5.5/common/lib/`.

Restart tomcat:

```
$ sudo /etc/init.d/tomcat5.5 restart
```

2.1.7. CLASSPATH environment variable setting

Specify current directory (`.`), JAR files in `/usr/local/axis/lib/` and `/usr/share/tomcat5.5/common/lib/` at CLASSPATH environment variable in `/etc/profile`.

Edit `/etc/profile`:

```
$ sudo vi /etc/profile
```

and, append the following settings at the end of the file.

```
export AXIS_HOME=/usr/local/axis/  
export CATALINA_HOME=/usr/share/tomcat5.5/  
CLASSPATH=.:$AXIS_HOME/lib/axis-ant.jar:$AXIS_HOME/lib/axis-schema.jar:$A  
XIS_HOME/lib/axis.jar:$AXIS_HOME/lib/commons-discovery-0.2.jar:$AXIS_HOME"  
lib/commons-logging-1.0.4.jar:$AXIS_HOME/lib/jaxrpc.jar:$AXIS_HOME/lib/log4j-1.  
2.8.jar:$AXIS_HOME/lib/saaj.jar:$AXIS_HOME/lib/wsdl4j-1.5.1.jar:$CATALINA_H  
OME"common/lib/activation.jar:$CATALINA_HOME"common/lib/mail.jar:$CATALIN  
A_HOME"common/lib/xmlsec-1.4.1.jar:$CATALINA_HOME"common/lib/postgresql-8.
```

```
1-409.jdbc3.jar
export CLASSPATH
```

Confirmation of the variable:

```
$ source /etc/profile
$ echo $CLASSPATH
./usr/local/axis/lib/axis-ant.jar:/usr/local/axis/lib/axis.jar:/usr/local/axis/lib/commons-discovery-0.2.jar:/usr/local/axis/lib/commons-logging-1.0.4.jar:/usr/local/axis/lib/jaxrpc.jar:/usr/local/axis/lib/log4j-1.2.8.jar:/usr/local/axis/lib/saaj.jar:/usr/local/axis/lib/wsdl4j-1.5.1.jar:/usr/share/tomcat5.5/common/lib/activation.jar:/usr/share/tomcat5.5/common/lib/mail.jar:/usr/share/tomcat5.5/common/lib/xmlsec-1.4.1.jar:/usr/share/tomcat5.5/common/lib/postgresql-8.1-409.jdbc3.jar
```

Confirm the specified files certainly exist. Version numbers sometimes are different.

Test of AxisClient execution:

```
$ java org.apache.axis.client.AdminClient
It is successful when this gives its manual.
```

2.2. Live E! system component Installation

Download Live E! server package from <http://live-e.hongo.wide.ad.jp/dist/>. This section describes the installation steps of a Live E! server. The installation involves settings of LIVEE_HOME and CLASSPATH environmental variables. The structure of the package (live-e-datamanager-x.x.x.zip) is:

```
live-e-datamanager-x.x.x
+ bin -- Daemon Controler
+ lib -- JAR Library
- live-e-datamanager-x.x.x.jar
+ conf -- Configurations
- livee_config.xml
- deploy_masterLiveE.wsdd
- deploy_slaveLiveE.wsdd
- undeploy_masterLiveE.wsdd
- undeploy_slaveLiveE.wsdd
```

- + schema -- Database Schema
 - live-e-datamanager-x.x.x.sql
- + log -- Output Logs
- + sample
 - profile_template.xml -- Template sensor profile

2.2.1. Package extraction and Variable settings

Extract the package as follows:

```
$ cd /usr/local/  
$ sudo unzip -----/live-e-datamanager-x.x.x.zip  
$ sudo ln -s live-e-datamanager-x.x.x livee
```

Change the permission of log directory:

```
$ sudo chown -R tomcat5.nogroup /usr/local/livee/log
```

LIVEE_HOME and CLASSPATH variable setting:

```
$ sudo vi /etc/profile  
insert,  
    export LIVEE_HOME=/usr/local/livee  
in the file, and append  
    /usr/local/livee/lib/live-e-datamanager-x.x.x.jar  
in the list of CLASSPATH definition.
```

2.2.2. Live E! Database Creation

DB user creation:

```
$ sudo su - postgres  
$ createuser <your login username>  
Shall the new role be the super user?(y/n) → choose “y”  
$ exit
```

Live E! DB creation:

```
$ createdb livee  
CREATE DATABASE
```

Live E! DB Schema Loading:

```
$ cd /usr/local/livee/schema
```



```
$ psql livee -f live-e-datamanager-x.x.x.sql
```

2.2.3. Live E! component installation in Axis

Create a file link to include live-e-datamanager into Axis as follows:

```
$ cd /var/lib/tomcat5.5/webapps/axis/WEB_INF/lib/
```

```
$ sudo ln /usr/local/livee/lib/live-e-datamanager-x.x.x.jar live-e-datamanager-x.x.x.jar
```

2.2.4. Live E! service deployment on Axis web service

The services that should be deployed are different, depending on the mode (master/slave) of the server. Be sure to deploy the correct service.

If the server works as a master server:

```
$ cd /usr/local/livee/conf/
```

```
$ java org.apache.axis.client.AdminClient -p 8180 deploy_masterLiveE.wsdd
```

If the server works as a slave server:

```
$ cd /usr/local/livee/conf/
```

```
$ java org.apache.axis.client.AdminClient -p 8180 deploy_slaveLiveE.wsdd
```

2.3. Live E! Server Configuration

Live E! server configuration is described in livee_config.xml. This section only gives how to configure neighbor topology information. Live E! manages a server community, which is connected over the Internet by tree-structured topology. Each server must be included into the community (by an upper site) and also can include other servers into the community (as lower sites). This section shows the configuration of neighbor topology information that implements the architecture. For the detail of server configuration, see Appendix A.

The configuration file is written in XML, and the neighbor topology information is located at /c:liveeConfiguration/c:neighbor/c:topology/a:neighbor.

Here,

```
xmlns:c="http://live-e.org/Configuration/2007/03/"
```

```
xmlns:a="http://live-e.org/Administrator/2007/03/"
```

The naming rule of Live E! server node is very similar to that of domain name system(DNS). The root server is identified by dot(.) and each server just under the root

has a country-code name basically: e.g., “jp.” and “tw.”. These servers have authority of operating their sub-trees and they can allocate names on their child servers on their own authority; e.g., “jp.” server can allocate “hoge.jp.” and “hoge.hoge.jp.” to their child servers. These names and corresponding servers and organizations are called “site” in Live E!. Each site can setup redundant servers(Section 5).

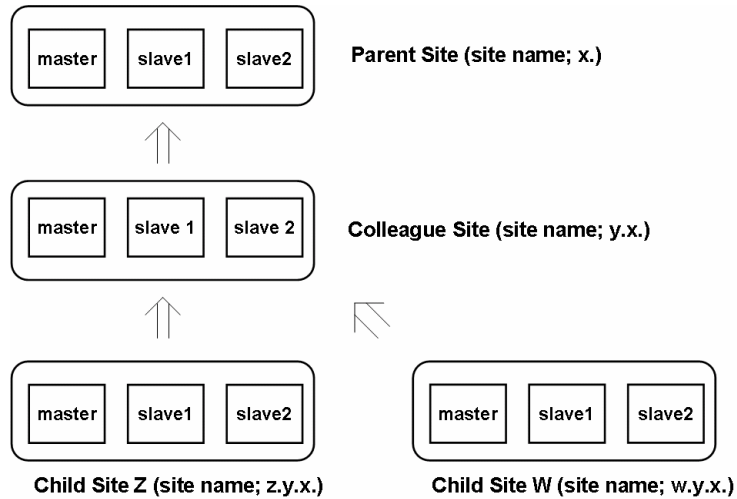


Figure 1: Neighbor Topology Example for site “y.x.”

The configuration of neighbor topology information for server “y.x.” in figure 1 is shown in figure 2.

```

<neighbor xmlns="http://live-e.org/Administrator/2007/03">
  <site type="parent" name="x.">
    <server name="Admin200703" url="http://master.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave1.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave2.---/axis/services/Admin200703" />
  </site>
  <site type="colleague" name="y.x.">
    <server name="Admin200703" url="http://master.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave1.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave2.---/axis/services/Admin200703" />
  </site>
  <site type="child" name="z.y.x.">
    <server name="Admin200703" url="http://master.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave1.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave2.---/axis/services/Admin200703" />
  </site>
  <site type="child" name="w.y.x.">
    <server name="Admin200703" url="http://master.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave1.---/axis/services/Admin200703" />
    <server name="Admin200703" url="http://slave2.---/axis/services/Admin200703" />
  </site>
</neighbor>

```

Figure 2: Configuration of Neighbor Topology in “y.x.”

Practically, when the server tree is constructed as in figure 3, the configurations at “fr.” and “fire04.fr.” can be set as figure 4 and figure 5 respectively.

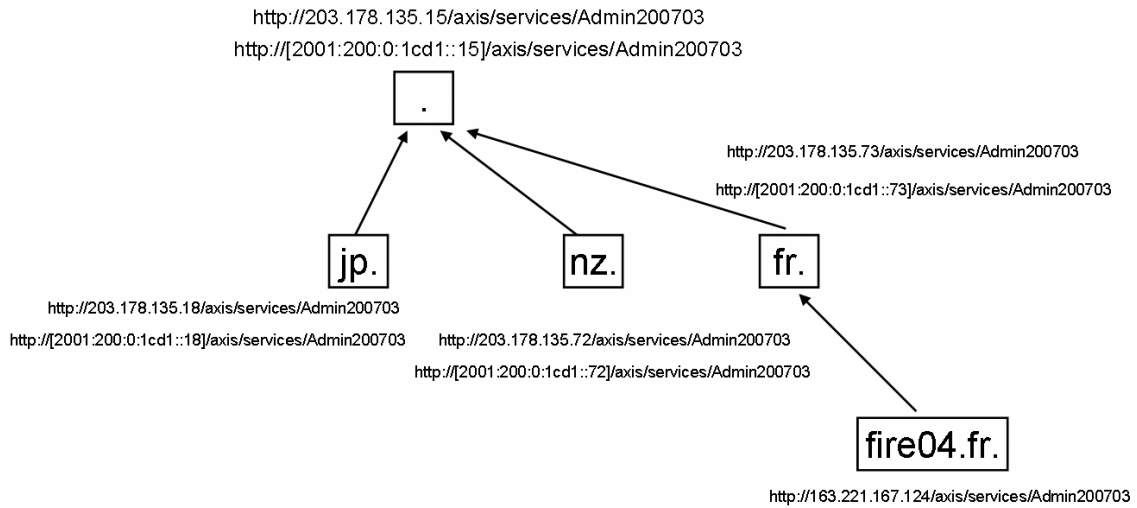


Figure 3: Example of practical server tree

```
<neighbor xmlns="http://live-e.org/Administrator/2007/03/">
  <site type="parent" name=".">
    <server name="Admin200703" url="http://[2001:200:0:1cd1::15]/axis/services/Admin200703" />
    <server name="Admin200703" url="http://203.178.135.15/axis/services/Admin200703" />
  </site>
  <site type="colleague" name="fr.">
    <server name="Admin200703" url="http://[2001:200:0:1cd1::73]/axis/services/Admin200703" />
    <server name="Admin200703" url="http://203.178.135.73/axis/services/Admin200703" />
  </site>
  <site type="child" name="fire04.fr.">
    <server name="Admin200703" url="http://163.221.167.124/axis/services/Admin200703" />
  </site>
</neighbor>
```

Figure 4: Neighbor Topology Configuration at “fr.”

```
<neighbor xmlns="http://live-e.org/Administrator/2007/03/">
  <site type="parent" name="fr.">
    <server name="Admin200703" url="http://[2001:200:0:1cd1::73]/axis/services/Admin200703" />
    <server name="Admin200703" url="http://203.178.135.73/axis/services/Admin200703" />
  </site>
  <site type="colleague" name="fire04.fr.">
    <server name="Admin200703" url="http://163.221.167.124/axis/services/Admin200703" />
  </site>
</neighbor>
```

Figure 5: Neighbor Topology Configuration at “fire04.fr.”

After the configuration, start “Administrator”

```
$ sudo /usr/local/livee/bin/administrator.sh start
```

Restart Tomcat

```
$ sudo /etc/init.d/tomcat5.5 restart
```

2.4. Server Test (Running Status)

```
$ java org.livee.test.TestAdmin200703
```

This command gives server status in XML format. Check that it does not give any error message.

2.5. Server Log

Server log will be put in /usr/local/livee/log/

- live-e.log --- Normal Log
- live-e-warning.log --- Warning Message
- live-e-error.log --- Error Message

Check live-e-warning.log and live-e-error.log to confirm that it is working properly. Any questions are welcomed at live-e[at]mri.co.jp. These server log messages can be transferred by e-mail, see Section 3.3.

Section 3: Server Operation

3.1. Sensor Registration

Live E! server operators must authorize sensors with their authority. This authorization can be performed by sensor accounting: i.e., sensor registration and sensor authentication. Live E! server system has a sensor accounting command to achieve that account-based authorization.

3.1.1. Sensor ID Format

. Combined Sensor ID

Live E! defines “combined sensor” as a packaged sensor of multiple sensors (e.g., temperature sensor and humidity sensor). Each combined sensor must have globally-unique ID. Live E! defines ID format that enables the global-uniqueness as follows:

CombinedSensorID ::= FQDN of the organization “/” sensorModel “/” free-format “/”

For example, a combined sensor ID of the sensor deployed at the room B206 of “Nara Institute of Science and Technology (NAIST)”, which sensor model is WM918, can be:

live-e.naist.jp/WM918/B206/

(FQDN of the organization / sensorModel / free-format /)

Note: Combined sensor ID must be ended with slash (/).

. Element Sensor ID

This ID should be allocated on every elemental sensor that can work as an independent sensor. The format should be:

ElementSensorID ::= CombinedSensorID sensorType

The current approved sensor types are listed here:

- Temperature
- Humidity
- RainFall
- DayRainFall
- WindDir
- WindSpeed
- Illuminance

For example, the element sensor ID for “temperature” sensor of live-e.naist.jp/WM918/B206/ is:

live-e.naist.jp/WM918/B206/Temperature

Note: element ID must NOT be ended with slash (/).

3.1.2. Sensor Accounting

To make an account for a new sensor, login to the server and use “Authorization” command as follows:

```
$ java org.livee.data.command.Authorization add -i hongo.wide.ad.jp/WM918/elab/ -p 00000000 -o “Hideya Ochiai” -m jo2lxq@hongo.wide.ad.jp
```

“add” means new sensor registration, -i specifies the combined sensor ID, -p specifies password, -o specifies owner name of the sensor, and -m specifies mail address of the sensor owner. For the detail of the command, see Appendix B.

3.1.3. Sensor Profile Registration

After accounting a sensor, profile of the sensor must be registered to the server by “ProfileManager” command or any other profile registration tools. A sensor profile gives additional information to its sensor: e.g., geographical address, vendor name, latitude and longitude. Sensor profile is also used for indexing sensor for applications. For profile format, see Appendix C.

ProfileManager command:

```
$ java org.livee.data.command.ProfileManager set -f FILEPATH -p 00000000
```

The template file is extracted at:

```
$LIVEE_HOME/sample/profile_template.xml
```

Profile management service is also provided by a web service. For the detail of the service, see Appendix D.

Note: SensorRegisterDX can register sensor profile via the Web service. This software is available at <http://live-e.naist.jp/SensorRegisterDX/>

3.1.4. Sensor Data Upload

Sensor data uploading service is provided by a web service (DataUpload200703). All the sensors must use this service to publish its observed sensor data.

The web service (DataUpload200703) has the following three remote methods for several upload granularities. See also Appendix E.

String uploadElement(String data)

String uploadCombined(String data)

String uploadCollection(String data)

Armadillo sensor hosts for WM918 / WXT510 which use the web service are available at <http://live-e.hongo.wide.ad.jp/DataUpload200703/>

3.2. Local Data Retrieval

```
$ java org.livee.data.command.LocalDataRetriever
```

Usage: java org.livee.data.command.LocalDataRetriever COMMAND [OPTIONS]

COMMAND:= getProfileSchema | getLatestDataAll
| getLatestData | getDataHourlyAggregated
| getDataDailyAggregated | getDataMonthlyAggregated
| getProfileAll | getProfile

OPTIONS:=(OPTION)* | e

OPTION:= -i ID | -s START_TIMESTAMP | -e END_TIMESTAMP
| -l LOCALE | -tz TIMEZONE

\$ java org.livee.data.command.CSVArchiver

Usage: java org.livee.data.command.CSVArchiver ID [OPTIONS]

OPTIONS:=(OPTION)* | e

OPTION:= -s START_TIMESTAMP | -e END_TIMESTAMP

3.3. Logging and Message Notification

Server log will be put in /usr/local/livee/log/

- live-e.log --- Normal Log
- live-e-warning.log --- Warning Message
- live-e-error.log --- Error Message

These messages can be transferred by e-mail. Figure 6 shows logging system configuration. The configuration is written at /liveeConfiguration/logging in the configuration file (livee_config.xml).

```
<logging>
  <smtpServer></smtpServer>
  <normal>
    <mail></mail>
    <file>live-e.log</file>
  </normal>
  <warning>
    <mail></mail>
    <file>live-e-warning.log</file>
  </warning>
  <error>
    <mail></mail>
    <file>live-e-error.log</file>
  </error>
</logging>
```

Figure 6: Logging System Configuration

To receive messages by e-mail, specify SMTP server in “smtpServer” and mail addresses in “mail” XML nodes. Then, restart “Administrator” and “Tomcat”.

Section 4: Global Sensor Search and Data Retrieval

Server installation and joining in the global Live E! server network enables to search and retrieve sensor data from the global community. Live E! server provides search and retrieval service interface by a web service (GlobalDataProvider200703). This service provides the following functionalities.

- Profile Schema Retrieval
- Sensor Profile Retrieval
- Sensor Data Retrieval
- Language Locale Selection
- Time Zone Selection
- Geographical Region Selection
- Aggregated Values (e.g., Maximum, Minimum and Average)
- Server Search

Please also refer to Appendix F, which describes the detail of the service.

Server administrator can also search and retrieve sensor from the server console.

```
$ java org.livee.data.command.GlobalDataResolver
```

Section 5: Redundant Services and Data for Robustness

Redundancy is necessary for improving service and data robustness. Live E! enables the redundancy with Master / Slave technique. Slave servers synchronizes their master server, and data search links, retrieval interfaces and data itself will be redundantly saved. An operational unit is called “site” in Live E!. Any sites can setup redundant servers in this way.

The following is the setting steps.

1. Slave server configuration
2. Service deployment
3. Binding with the Live E! server tree

5.1. Slave Server Configuration

5.1.1. Mode setting

In the configuration file(livee_config.xml), alter

```
<mode>master</mode>
```

with

```
<mode>slave</mode>
```

5.1.2. Master server setting

```
<slave>
  <masterAdminURL>http://live-e2.hongo.wide.ad.jp/axis/services/Admin200703 </masterAdminURL>
  . . .
</slave>
```

Alter the underlined URL with your master's Admin200703 service URL.

5.2. Service Deployment

Confirm the existence of live-e-datamanager-x.x.x.jar in WEB_INF/lib/ directory in Axis on Tomcat, and deploy services for slave settings as follows:

```
$ cd /usr/local/livee
```

```
$ cd conf
```

```
$ java org.apache.axis.client.AdminClient deploy_slaveLiveE.wsdd
```

Note: If master services are already deployed,

```
$ java org.apache.axis.client.AdminClient undeploy_masterLiveE.wsdd
```

```
$ java org.apache.axis.client.AdminClient deploy_slaveLiveE.wsdd
```

Then, reboot tomcat to enable.

5.3. Binding with the Live E! server tree

The operator must have the parent site registered the redundant service list. In neighbor topology configuration, redundant services must be set at 'site type="colleague"' as follows:

```
<site type="colleague" name="jp.">
  <server service="Admin200703" url="http://master.domain.com/axis/services/Admin200703" />
  <server service="Admin200703" url="http://slave0.domain.com/axis/services/Admin200703" />
  <server service="Admin200703" url="http://slave1.domain.com/axis/services/Admin200703" />
</site>
```

When a child site operator notifies their topology update, configure the topology

information at the corresponding site 'type="child" name="xxx".

Section 6: Sensor profile consistency management with schema

Live E! system has a single profile schema maintained by Live E! authority to improve sensor profile consistency and to index sensors using their profile. Live E! authority publishes the schema at the root server of the tree, and the schema will be automatically disseminated to all the server on down links of the tree. Using the schema, profiles are verified periodically or at the registration, and if mismatches are found, warning messages will be posted.

The following is the detail of the profile schema.

Structure

```
<profileSchema xmlns="http://live-e.org/Schema/2007/03/">
  <schema name="location" class="combined" type="string" value=".*" multilanguage="true" />
  <schema name="latitude" class="combined|element" type="float" />
  . . .
</profileSchema>
```

Schema attributes and the meanings

Attribute	Meaning
name	The name of the attribute in profile.
class	The layer of the attribute to appear in profile (regular expression). collection -- combined sensor collection layer combined -- combined sensor layer element -- element sensor layer value -- data layer
type	The type of the attribute value in profile. boolean, integer, float, time, string
value	The allowed values of the attribute (regular expression).
multilanguage	Enable / Disable multilanguage extension When multilanguage="true", this attribute can support multi-languages with the following rule. Language_Dependent_AttrName ::= AttrName "_" LanguageCode

	See Appendix G for the available LanguageCode e.g.) location_jpn, location_eng, location_tha, location_fre
delegation	Delegate to external application domain. (Reserved)
description	The semantics of the attribute. (Reserved)

Section 7: Requirements for developing and deploying new sensors

To develop a new sensor and to use it in Live E! environment;

1. Create a system that uploads data using the web service (DataUpload200703).
2. Apply new sensor information to Live E! authority

For the first requirement, a developer can download the tool from

<http://live-e.hongo.wide.ad.jp/DataUpload200703/>

For the second requirement, please contact to live-e[at]mri.co.jp with information about the new type of sensor: i.e., vendor, model, and measurement type. After a small discussion, sensorVendor, sensorModel, and sensorType will be approved and determined in Live E!.

Appendix A: Configuration (livee_config.xml)

livee_config.xml

```
<liveeConfiguration xmlns="http://live-e.org/Configuration/2007/03/">

<mode>master</mode>
<dbAccess>jdbc:postgresql:livee</dbAccess>

<master>
  <aggregator>
    <start>2007-07-01T00:00:00.0000000+09:00</start>
    <interval>300</interval>
    <hourAggregator>
      <thresholdTime>70</thresholdTime>
      <thresholdTimeBase>minute</thresholdTimeBase>
    </hourAggregator>
    <dayAggregator>
      <thresholdTime>1</thresholdTime>
      <thresholdTimeBase>day</thresholdTimeBase>
    </dayAggregator>
    <monthAggregator>
      <thresholdTime>1</thresholdTime>
      <thresholdTimeBase>month</thresholdTimeBase>
    </monthAggregator>
  </aggregator>

  <profileSchemaLoader>
    <interval>3600</interval>
  </profileSchemaLoader>

  <childProfileMerger>
    <interval>3600</interval>
    <cacheValidityTime>7200</cacheValidityTime>
  </childProfileMerger>

  <dataManagerProfileMerger>
    <interval>1200</interval>
  </dataManagerProfileMerger>

  <profileChecker>
    <interval>600</interval>
    <errOutputLevel>warning</errOutputLevel>
  </profileChecker>

  <sensorLivingChecker>
    <interval>3600</interval>
    <thresholdTime>3</thresholdTime>
    <thresholdTimeBase>day</thresholdTimeBase>
    <errOutputLevel>warning</errOutputLevel>
  </sensorLivingChecker>
</master>

<slave>
  <masterAdminURL>http://live-e2.hongo.wide.ad.jp/axis/services/Admin200703</masterAdminURL>
  <copyAdmin>
    <interval>600</interval>
  </copyAdmin>
  <copyDataManager>
</slave>

<masterURL>http://live-e2.hongo.wide.ad.jp/axis/services/DataManagerReplication200703</masterURL>
  <latestDataUpdate>
    <interval>60</interval>
  </latestDataUpdate>
  <profileUpdate>
    <start>2007-07-01T00:00:00.0000000+09:00</start>
    <interval>300</interval>
    <thresholdTime>600</thresholdTime>
    <thresholdTimeBase>second</thresholdTimeBase>
  </profileUpdate>
</masterURL>
```

```

    <scopeTime>300</scopeTime>
    <scopeTimeBase>second</scopeTimeBase>
  </profileUpdate>
  <archiveDataUpdate>
    <start>2007-07-01T00:00:00.0000000+09:00</start>
    <interval>300</interval>
    <thresholdTime>600</thresholdTime>
    <thresholdTimeBase>second</thresholdTimeBase>
    <scopeTime>300</scopeTime>
    <scopeTimeBase>second</scopeTimeBase>
  </archiveDataUpdate>
</copyDataManager>
</slave>

<neighbor>
  <topology>
    <neighbor xmlns="http://live-e.org/Administrator/2007/03/">
      <site type="parent" name="live-e.org">
        <server service="Admin200703" url="http://-----/axis/services/Admin200703" />
      </site>
      <site type="colleague" name="thai.live-e.org">
        <server service="Admin200703" url="http://----/axis/services/Admin200703" />
      </site>
      <site type="child" name="ku.thai.live-e.org">
        <server service="Admin200703" url="http://----/axis/services/Admin200703" />
      </site>
      <site type="child" name="au.thai.live-e.org">
        <server service="Admin200703" url="http://----/axis/services/Admin200703" />
      </site>
    </neighbor>
  </topology>
</linkStatusChecker>
  <interval>300</interval>
</linkStatusChecker>
<sharedTopologyLoader>
  <interval>300</interval>
</sharedTopologyLoader>
</neighbor>

<profileSchema>
  <interval>300</interval>
  <schema>
    <adminSchema xmlns="http://live-e.org/Administrator/2007/03/">
      <locales xmlns="http://live-e.org/Schema/2007/03/">
        <locale code="jpn" name="Japanese" />
        <locale code="eng" name="English" />
        <locale code="fre" name="French" />
        <locale code="ger" name="German" />
        <locale code="kor" name="Korean" />
        <locale code="chi" name="Chinese" />
        <locale code="tha" name="Thai" />
      </locales>
      <profileSchema xmlns="http://live-e.org/Schema/2007/03/" >
        <!-- type is one of boolean, integer, float, time, string -->
        <schema name="location" type="string" value=".*" multilanguage="true" class="combined" />
        <schema name="address" type="string" value=".*" multilanguage="true" class="combined" />
        <schema name="latitude" type="float" value=".*" class="combined|element" />
        <schema name="longitude" type="float" value=".*" class="combined|element" />
        <schema name="altitude" type="float" value=".*" class="combined|element" />
        <schema name="gAltitude" type="float" value=".*" class="combined|element" />
        <schema name="sensorVendor" type="string"
value="Vaisala|AmbientWeather|Davis|Ubiteq|MatsushitaDenko|TriState"
class="combined|element" />
        <schema name="sensorModel" type="string"
value="WXT510|WM918|VantagePRO|WSN-100X|FS-Va-01|PICNICv12" class="combined|element" />
        <schema name="sensorType" type="string"
value="Temperature|Humidity|Pressure|DayRainFall|RainFall|WindSpeed|WindDir|CO2"
class="element" />
        <schema name="accuracy" type="float" value=".*" class="element" />
        <schema name="error" type="float" value=".*" class="element" />
        <schema name="time" type="time" value=".*" class="value" />
      </profileSchema>
    </adminSchema>
  </schema>
</profileSchema>

```

```
</adminSchema>
</schema>
</profileSchema>

<search>
  <queryResolver>
    <cacheValidityTime>600</cacheValidityTime>
    <cacheSize>100</cacheSize>
  </queryResolver>
  <retrieveManager>
    <cacheValidityTime>60</cacheValidityTime>
    <cachePredicate>.*</cachePredicate>
    <cacheSize>10</cacheSize>
  </retrieveManager>
</search>

<restriction>
  <maxManageableSensorCount>
    <warning>128</warning>
    <error>256</error>
  </maxManageableSensorCount>
  <maxSensorCollectionCount>
    <warning>2048</warning>
    <error>8192</error>
  </maxSensorCollectionCount>
  <maxValueCollectionCount>
    <warning>16384</warning>
    <error>32768</error>
  </maxValueCollectionCount>
  <replication>
    <maxAggregationCollectionCount>
      <warning>1024</warning>
      <error>2048</error>
    </maxAggregationCollectionCount>
    <maxProfileCombinedArchiveCount>
      <warning>2048</warning>
      <error>4096</error>
    </maxProfileCombinedArchiveCount>
    <maxProfileElementArchiveCount>
      <warning>16384</warning>
      <error>32768</error>
    </maxProfileElementArchiveCount>
    <maxRawArchiveCollectionCount>
      <warning>65536</warning>
      <error>131072</error>
    </maxRawArchiveCollectionCount>
    <maxRawLatestCollectionCount>
      <warning>1024</warning>
      <error>2048</error>
    </maxRawLatestCollectionCount>
    <maxSensorCombinedCollectionCount>
      <warning>128</warning>
      <error>256</error>
    </maxSensorCombinedCollectionCount>
    <maxSensorElementCollectionCount>
      <warning>1024</warning>
      <error>2048</error>
    </maxSensorElementCollectionCount>
  </replication>
</restriction>

<queryService>
  <service>
    <name>Admin200703</name>
    <dirPath>/axis/services/Admin200703</dirPath>
  </service>

  <service>
    <name>DataManagerReplication200703</name>
    <dirPath>/axis/services/DataManagerReplication200703</dirPath>
  </service>
</queryService>
```

```
<service>
  <name>DataProvider200703</name>
  <dirPath>/axis/services/DataProvider200703</dirPath>
</service>

<service>
  <name>DataUpload200703</name>
  <dirPath>/axis/services/DataUpload200703</dirPath>
</service>

<service>
  <name>GlobalDataProvider200703</name>
  <dirPath>/axis/services/GlobalDataProvider200703</dirPath>
</service>

<service>
  <name>ProfileManagement200703</name>
  <dirPath>/axis/services/ProfileManagement200703</dirPath>
</service>

</queryService>

<logging>
  <smtpServer></smtpServer>
  <normal>
    <mail></mail>
    <file>live-e.log</file>
  </normal>
  <warning>
    <mail></mail>
    <file>live-e-warning.log</file>
  </warning>
  <error>
    <mail></mail>
    <file>live-e-error.log</file>
  </error>
</logging>

</liveConfiguration>
```

Appendix B: Sensor Administration Command

Use “Authorization” command to register / leave sensors.

```
$ java org.livee.data.command.Authorization
```

```
Usage: java org.livee.data.command.Authorization COMMAND [OPTIONS]
```

```
COMMAND ::= list | add | update | close | restore
```

```
OPTIONS ::= (OPTION)* |
```

```
OPTION ::= -i ID | -p PASSWORD | -o OWNER_NAME | -m OWNER_MAIL_ADDRESS
```

Five sub commands are provided.

- list (list the registered sensors)
- add (new sensor registration)
 - Required: ID, Password, OwnerName, OwnerMailAddress
- update (sensor entity update including the password)
 - Required: ID; Optional: Password, OwnerName, OwnerMailAddress
- close (stop and leave the sensor from the external link)
 - Required: ID
- restore
 - Required: ID

Appendix D: Profile Management Service

Live E! web service for sensor profile registration and modification.

Web Service: ProfileManagement200703

```
String getAvailableLocales();  
String getProfileSchema();  
String getProfileAll(String tz,String lang);  
String getProfile(String id);  
String setProfile(String data);  
String updatePassword(String id, String old_pass, String new_pass);
```

String getAvailableLocales();

This method gives the list of available language locales.

String getProfileSchema();

This method gives the system profile schema (see Section 6).

String getProfileAll(String tz,String lang);

This method gives the whole sensor profile registered in the server with time zone(tz) and in language (lang). This method can be used for obtaining registered sensor list.

String getProfile(String id);

This method retrieves a registered sensor profile specified by “id”. The retrieved profile format differs from that of DataProvider200703 in that this method gives multi-language profile at the same time (Appendix C). Users can use this method for modifying and updating sensor profile with setProfile method.

String setProfile(String profile);

This method registers sensor profile formatted as in Appendix C.

String updatePassword(String id,String old_passwd,String new_passwd);

This method can be used for updating sensor access password. “old_passwd” and “new_password” must be encoded in BASE64.

Appendix E: Sensor Data Upload Service

Live E! web service for uploading sensor data to the server.

Web Service: DataUpload200703

- String uploadElement(String data)
- String uploadCombined(String data)
- String uploadCollection(String data)

String uploadElement(String xml)

This method is provided for uploading data by sensor element. Multiple values are allowed. Sensor access password must be encoded in BASE64.

```
<?xml version="1.0" encoding="UTF-8" ?>
<sensor          id="live-e.naist.jp/WM918/Temperature"          authorization="MDAwMDAwMDA="
xmlns="http://live-e.org/DataType/2007/03/" >
  <value time="2007-06-27T00:00:00.0000000+09:00">25.6</value>
  <value time="2007-06-27T00:10:00.0000000+09:00">25.5</value>
  ...
</sensor>
```

String uploadCombined(String xml)

This method is provided for uploading data by sensor unit (combined sensor). Multiple values are allowed. Sensor access password must be encoded in BASE64.

```
<?xml version="1.0" encoding="UTF-8" ?>
<sensorGroup    authorization="MDAwMDAwMDA="    class="combined"    id="live-e.naist.jp/WM918/B206/"
xmlns="http://live-e.org/DataType/2007/03/">
  <sensor id="live-e.naist.jp/WM918/B206/Temperature">
    <value time="2007-06-27T00:00:00.0000000+09:00">25.5</value>
    <value time="2007-06-27T00:10:00.0000000+09:00">25.3</value>
    ...
  </sensor>
  <sensor id="live-e.naist.jp/WM918/B206/Humidity" >
    <value time="2007-06-27T00:00:00.0000000+09:00">56.5</value>
    <value time="2007-06-27T00:10:00.0000000+09:00">56.8</value>
    ...
  </sensor>
  <sensor id="live-e.naist.jp/WM918/B206/Pressure" >
    <value time="2007-06-27T00:00:00.0000000+09:00">1003</value>
    <value time="2007-06-27T00:10:00.0000000+09:00">1006</value>
    ...
  </sensor>
  <sensor id="live-e.naist.jp/WM918/B206/WindDir" >
    <value time="2007-06-27T00:00:00.0000000+09:00">352</value>
    <value time="2007-06-27T00:00:20.0000000+09:00">132</value>
    ...
  </sensor>
  <sensor id="live-e.naist.jp/WM918/B206/WindSpeed" >
    <value time="2007-06-27T00:00:00.0000000+09:00">2.4</value>
```

```
<value time="2007-06-27T00:00:20.0000000+09:00">3.5</value>
...
</sensor>
<sensor id="live-e.naist.jp/WM918/B206/RainFall" >
  <value time="2007-06-27T00:00:00.0000000+09:00">0.0</value>
  <value time="2007-06-27T00:10:00.0000000+09:00">1.0</value>
  ...
</sensor>
</sensorGroup>
```

String uploadCollection(String xml)

This method is provided for uploading multiple combined sensor data at the same time in a transaction. Sensor access passwords must be encoded in BASE64 and must be put on each sensor field.

```
<?xml version="1.0" encoding="UTF-8" ?>
<sensorGroup class="collection" xmlns="http://live-e.org/DataType/2007/03/">
  <sensorGroup authorization="MDAwMDAwMDA=" class="combined" id="live-e.naist.jp/WM918/B206/" >
    ...
  </sensorGroup>
  <sensorGroup authorization="MDAwMDAwMDA=" class="combined" id="hongo.wide.ad.jp/WM918/elab/" >
    ...
  </sensorGroup>
  <sensorGroup authorization="MDAwMDAwMDA=" class="combined" id="im.unl.dendai.ac.jp/WM918/roof/" >
    ...
  </sensorGroup>
  ...
</sensorGroup>
```

Appendix F: Global Sensor Data Search/Retrieval

Live E! web service for searching and retrieving sensor data in the global Live E! network.

Web Service: GlobalDataProvider200703

```
String getProfileSchema()  
String getProfileAll(String tz,String lang)  
String getProfile(String id,String tz,String lang)  
String getProfileByAreaRect(double north,double south,double east,double west, String tz,String lang);  
String getArchiveProfile(String id,String start,String end,String tz,String lang)  
String getLatestDataAll(String tz,String lang)  
String getLatestData(String id,String tz,String lang)  
String getLatestDataByAreaRect(double north,double south,double east,double west,String tz,String lang);  
String getArchiveCombinedData(String id,String start,String end,String tz,String lang)  
String getDataHourlyAggregated(String id,String start,String end,String tz,String lang)  
String getDataDailyAggregated(String id,String start,String end,String tz,String lang)  
String getDataMonthlyAggregated(String id,String start,String end,String tz,String lang)  
String search(String query)
```

String getProfileSchema();

This method gives the system sensor profile, see Section 6.

String getProfileAll(String tz,String lang);

This method gives all the sensor profile globally with time zone (tz) in language (lang).

```
<sensorGroup class="collection" xmlns="http://live-e.org/DataType/2007/03/">  
  <sensorGroup class="combined" id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65"  
  longitude="139.345" location="江崎研究室" sensorVendor="AmbientWeather"  
  sensorModel="WM918" . . . >  
    <sensor id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature" />  
    <sensor id="hongo.wide.ad.jp/WM918/elab/Humidity" sensorType="Humidity" />  
    <sensor id="hongo.wide.ad.jp/WM918/elab/Pressure" sensorType="Pressure" />  
    . . . sensor . . .  
  </sensorGroup>  
  . . . sensorGroup class="combined" . . .  
</sensorGroup>
```

E.g., `getProfileAll("JST", "Japanese");`

String getProfile(String id,String tz,String lang);

This method retrieves and gives specified sensor profile by “id” with time zone(tz) in language (lang).

```
<sensorGroup class="combined" id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65"
longitude="139.345" location="Esaki Laboratory" sensorVendor="AmbientWeather"
sensorModel="WM918" . . . xmlns="http://live-e.org/DataType/2007/03/" >
  <sensor id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature" />
  <sensor id="hongo.wide.ad.jp/WM918/elab/Humidity" sensorType="Humidity" />
  <sensor id="hongo.wide.ad.jp/WM918/elab/Pressure" sensorType="Pressure" />
  . . . sensor の繰り返し . . .
</sensorGroup>
```

E.g., `getProfile("hongo.wide.ad.jp/WM918/elab/","JST","English");`

String getProfileByAreaRect(double north,double south,double east,double west, String tz,String lang);

This method retrieves sensor profile with geographical region specification [south,north]x[west,east]. North Pole: +90, South Pole: -90, 180 ° E: +180, and 180 ° W: -180.

String getArchiveProfile(String id, String start, String end, String tz, String lang)

This method retrieves sensor profile update logs with time specification[start,end]. “start” and “end” must be formatted with W3CTimestamp expression.

```
<sensorGroup class="collection" xmlns="http://live-e.org/DataType/2007/03/">
  <sensorGroup class="combined"
    created="2007-07-25T00:52:55.00000000+09:00"
    expired="2007-07-25T06:09:59.00000000+09:00"
    id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65" longitude="139.345"
    location="江崎研究室"
    sensorVendor="AmbientWeather" sensorModel="WM918" . . . >
    <sensor created="2007-07-25T00:52:55.00000000+09:00"
      expired="2007-07-25T06:09:59.00000000+09:00"
      id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature" />
    . . . sensor . . .
  </sensorGroup>
  <sensorGroup class="combined"
    created="2007-07-25T06:09:59.00000000+09:00"
    expired="2007-07-30T06:30:58.00000000+09:00"
    id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65" longitude="139.345"
    location="江崎研究室(工学部 2 号館)" sensorVendor="AmbientWeather"
    sensorModel="WM918" . . . >
```

```

<sensor created="2007-07-25T06:09:59.00000000+09:00"
        expired="2007-07-30T06:30:58.00000000+09:00"
        id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature" />
    . . . sensor . . .
</sensorGroup>
. . . sensorGroup class="combined" . . .
</sensorGroup>

```

E.g., `getArchiveProfile("hongo.wide.ad.jp/WM918/elab/",
"2007-01-01T00:00:00.00000000+09:00",
"2007-08-01T00:00:00.00000000+09:00",
"JST","Japanese")`

String getLatestDataAll(String tz,String lang)

This method gives all the latest sensor data with time zone (tz) in language (lang).

```

<sensorGroup class="collection" xmlns="http://live-e.org/DataType/2007/03/">
  <sensorGroup class="combined" id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65"
longitude="139.345" location=" 江崎研究室 " sensorVendor="AmbientWeather"
sensorModel="WM918" . . . >
    <sensor id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature">
      <value time="2007-07-30T12:34:23.00000000+09:00">26.7</value>
    </sensor>
    <sensor id="hongo.wide.ad.jp/WM918/elab/Humidity" sensorType="Humidity">
      <value time="2007-07-30T12:34:23.00000000+09:00">57.2</value>
    </sensor>
    . . . sensor . . .
  </sensorGroup>
  . . . sensorGroup class="combined" . . .
</sensorGroup>

```

E.g., `getLatestDataAll("JST","Japanese");`

String getLatestData(String id,String tz,String lang)

This method gives the latest sensor data of specified sensor (id) with time zone (tz) in language (lang).

String getLatestDataByAreaRect(double north, double south, double east, double west, String tz, String lang)

This method retrieves the latest sensor data with geographical region specification [south,north]x[west,east]. North Pole: +90, South Pole: -90, 180 ° E: +180, and 180 ° W: -180.

String getArchiveCombinedData(String id, String start, String end, String tz, String lang)

This method retrieves raw sensor data archive with time specification[start,end]. “start” and “end” must be formatted with W3CTimestamp expression.

```
<sensorGroup class="combined" id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65"
longitude="139.345" location=" 江崎研究室 " sensorVendor="AmbientWeather"
sensorModel="WM918" . . . xmlns="http://live-e.org/DataType/2007/03/">
  <sensor id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature">
    <value time="2007-07-30T00:00:00.0000000+09:00">26.7</value>
    <value time="2007-07-30T00:01:00.0000000+09:00">26.7</value>
    <value time="2007-07-30T00:02:00.0000000+09:00">26.8</value>
    <value time="2007-07-30T00:03:00.0000000+09:00">26.8</value>
    . . . value . . .
  </sensor>
  . . . sensor . . .
</sensorGroup>
```

E.g., getArchiveCombinedData("hongo.wide.ad.jp/WM918/elab/",
"2007-07-30T00:00:00.0000000+09:00",
"2007-07-30T01:00:00.0000000+09:00",
"JST","Japanese")

String getDataHourlyAggregated(String id, String start, String end, String tz, String lang)

This method retrieves hourly aggregated sensor data with time specification[start,end]. “start” and “end” must be formatted with W3CTimestamp expression. Aggregated values are maximum, minimum and average values in a region.

```
<sensorGroup class="combined" id="hongo.wide.ad.jp/WM918/elab/" latitude="35.65"
longitude="139.345" location=" 江崎研究室 " sensorVendor="AmbientWeather"
sensorModel="WM918" . . . xmlns="http://live-e.org/DataType/2007/03/">
  <sensor id="hongo.wide.ad.jp/WM918/elab/Temperature" sensorType="Temperature">
    <aggValue aggType="avg" timeScope="hour"
      time="2007-07-28T06:00:00.0000000+09:00" >26.7</aggValue>
    <aggValue aggType="max" timeScope="hour"
      time="2007-07-28T06:00:00.0000000+09:00" >27.3</aggValue>
    <aggValue aggType="min" timeScope="hour"
      time="2007-07-28T06:00:00.0000000+09:00" >25.5</aggValue>
    <aggValue aggType="sum" timeScope="hour"
      time="2007-07-28T06:00:00.0000000+09:00" >1602</aggValue>
```



```
<aggValue aggType="count" timeScope="hour"
    time="2007-07-28T06:00:00.0000000+09:00" >60</aggValue>
    . . . aggValue . . .
</sensor>
    . . . sensor . . .
</sensorGroup>
```

```
E.g., getDataHourlyAggregated("hongo.wide.ad.jp/WM918/elab/",
    "2007-07-28T06:00:00.0000000+09:00",
    "2007-07-28T09:00:00.0000000+09:00",
    "JST","Japanese")
```

String getDataDailyAggregated(String id, String start, String end, String tz, String lang)

This method retrieves daily aggregated sensor data with time specification[start,end]. "start" and "end" must be formatted with W3CTimestamp expression. Aggregated values are maximum, minimum and average values in a region.

String getDataMonthlyAggregated(String id, String start, String end, String tz, String lang)

This method retrieves monthly aggregated sensor data with time specification[start,end]. "start" and "end" must be formatted with W3CTimestamp expression. Aggregated values are maximum, minimum and average values in a region.

String search(String query)

This method searches servers that match "query".

Appendix G: Available Language Locales

Available language locales in August, 2007.

These are based on ISO639-2.

http://www.loc.gov/standards/iso639-2/php/code_list.php

Language Code	Spelling
jpn	Japanese
eng	English
fre	French
ger	German
kor	Korean
chi	Chinese
tha	Thai

Live E! is going to append locales in the near future.

Appendix H: Available Time Zones

Etc/GMT+12	America/Chihuahua	US/Central
Etc/GMT+11	America/Dawson_Creek	America/Bogota
MIT	America/Denver	America/Cayman
Pacific/Apia	America/Edmonton	America/Coral_Harbour
Pacific/Midway	America/Hermosillo	America/Detroit
Pacific/Niue	America/Inuvik	America/Eirunepe
Pacific/Pago_Pago	America/Mazatlan	America/Fort_Wayne
Pacific/Samoa	America/Phoenix	America/Grand_Turk
US/Samoa	America/Shiprock	America/Guayaquil
America/Adak	America/Yellowknife	America/Havana
America/Atka	Canada/Mountain	America/Indiana/Indianapolis
Etc/GMT+10	Etc/GMT+7	America/Indiana/Knox
HST	MST	America/Indiana/Marengo
Pacific/Fakaofu	MST7MDT	America/Indiana/Vevay
Pacific/Honolulu	Mexico/BajaSur	America/Indianapolis
Pacific/Johnston	Navajo	America/Iqaluit
Pacific/Rarotonga	PNT	America/Jamaica
Pacific/Tahiti	SystemV/MST7	America/Kentucky/Louisville
SystemV/HST10	SystemV/MST7MDT	America/Kentucky/Monticello
US/Aleutian	US/Arizona	America/Knox_IN
US/Hawaii	US/Mountain	America/Lima
Pacific/Marquesas	America/Belize	America/Louisville
AST	America/Cancun	America/Montreal
America/Anchorage	America/Chicago	America/Nassau
America/Juneau	America/Costa_Rica	America/New_York
America/Nome	America/El_Salvador	America/Nipigon
America/Yakutat	America/Guatemala	America/Panama
Etc/GMT+9	America/Managua	America/Pangnirtung
Pacific/Gambier	America/Menominee	America/Port-au-Prince
SystemV/YST9	America/Merida	America/Porto_Acre
SystemV/YST9YDT	America/Mexico_City	America/Rio_Branco
US/Alaska	America/Monterrey	America/Thunder_Bay
America/Dawson	America/North_Dakota/Center	America/Toronto
America/Ensenada	America/Rainy_River	Brazil/Acre
America/Los_Angeles	America/Rankin_Inlet	Canada/Eastern
America/Tijuana	America/Regina	Cuba
America/Vancouver	America/Swift_Current	EST
America/Whitehorse	America/Tegucigalpa	EST5EDT
Canada/Pacific	America/Winnipeg	Etc/GMT+5
Canada/Yukon	CST	IET
Etc/GMT+8	CST6CDT	Jamaica
Mexico/BajaNorte	Canada/Central	SystemV/EST5
PST	Canada/East-Saskatchewan	SystemV/EST5EDT
PST8PDT	Canada/Saskatchewan	US/East-Indiana
Pacific/Pitcairn	Chile/EasterIsland	US/Eastern
SystemV/PST8	Etc/GMT+6	US/Indiana-Starke
SystemV/PST8PDT	Mexico/General	US/Michigan
US/Pacific	Pacific/Easter	America/Anguilla
US/Pacific-New	Pacific/Galapagos	America/Antigua
America/Boise	SystemV/CST6	America/Aruba
America/Cambridge_Bay	SystemV/CST6CDT	

America/Asuncion	America/Argentina/Rio_Gallegos	Atlantic/Reykjavik
America/Barbados	America/Argentina/San_Juan	Atlantic/St_Helena
America/Boa_Vista	America/Argentina/Tucuman	Eire
America/Campo_Grande	America/Argentina/Ushuaia	Etc/GMT
America/Caracas	America/Bahia	Etc/GMT+0
America/Cuiaba	America/Belem	Etc/GMT-0
America/Curacao	America/Buenos_Aires	Etc/GMT0
America/Dominica	America/Catamarca	Etc/Greenwich
America/Glace_Bay	America/Cayenne	Etc/UCT
America/Goose_Bay	America/Cordoba	Etc/UTC
America/Grenada	America/Fortaleza	Etc/Universal
America/Guadeloupe	America/Godthab	Etc/Zulu
America/Guyana	America/Jujuy	Europe/Belfast
America/Halifax	America/Maceio	Europe/Dublin
America/La_Paz	America/Mendoza	Europe/Lisbon
America/Manaus	America/Miquelon	Europe/London
America/Martinique	America/Montevideo	GB
America/Montserrat	America/Paramaribo	GB-Eire
America/Port_of_Spain	America/Recife	GMT
America/Porto_Velho	America/Rosario	GMT0
America/Puerto_Rico	America/Sao_Paulo	Greenwich
America/Santiago	Antarctica/Rothera	Iceland
America/Santo_Domingo	BET	Portugal
America/St_Kitts	Brazil/East	UCT
America/St_Lucia	Etc/GMT+3	UTC
America/St_Thomas	America/Noronha	Universal
America/St_Vincent	Atlantic/South_Georgia	WET
America/Thule	Brazil/DeNoronha	Zulu
America/Tortola	Etc/GMT+2	Africa/Algiers
America/Virgin	America/Scoresbysund	Africa/Bangui
Antarctica/Palmer	Atlantic/Azores	Africa/Brazzaville
Atlantic/Bermuda	Atlantic/Cape_Verde	Africa/Ceuta
Atlantic/Stanley	Etc/GMT+1	Africa/Douala
Brazil/West	Africa/Abidjan	Africa/Kinshasa
Canada/Atlantic	Africa/Accra	Africa/Lagos
Chile/Continental	Africa/Bamako	Africa/Libreville
Etc/GMT+4	Africa/Banjul	Africa/Luanda
PRT	Africa/Bissau	Africa/Malabo
SystemV/AST4	Africa/Casablanca	Africa/Ndjamena
SystemV/AST4ADT	Africa/Conakry	Africa/Niamey
America/St_Johns	Africa/Dakar	Africa/Porto-Novo
CNT	Africa/El_Aaiun	Africa/Tunis
Canada/Newfoundland	Africa/Freetown	Africa/Windhoek
AGT	Africa/Lome	Arctic/Longyearbyen
America/Araguaina	Africa/Monrovia	Atlantic/Jan_Mayen
America/Argentina/Buenos_Aires	Africa/Nouakchott	CET
America/Argentina/Catamarca	Africa/Ouagadougou	ECT
America/Argentina/ComodRivadavia	Africa/Sao_Tome	Etc/GMT-1
America/Argentina/Cordoba	Africa/Timbuktu	Europe/Amsterdam
America/Argentina/Jujuy	America/Danmarkshavn	Europe/Andorra
America/Argentina/La_Rioja	Atlantic/Canary	Europe/Belgrade
America/Argentina/Mendoza	Atlantic/Faeroe	Europe/Berlin
	Atlantic/Madeira	Europe/Bratislava
		Europe/Brussels

Europe/Budapest	Europe/Helsinki	Europe/Samara
Europe/Copenhagen	Europe/Istanbul	Indian/Mahe
Europe/Gibraltar	Europe/Kaliningrad	Indian/Mauritius
Europe/Ljubljana	Europe/Kiev	Indian/Reunion
Europe/Luxembourg	Europe/Mariehamn	NET
Europe/Madrid	Europe/Minsk	Asia/Kabul
Europe/Malta	Europe/Nicosia	Asia/Aqtou
Europe/Monaco	Europe/Riga	Asia/Aqtobe
Europe/Oslo	Europe/Simferopol	Asia/Ashgabat
Europe/Paris	Europe/Sofia	Asia/Ashkhabad
Europe/Prague	Europe/Tallinn	Asia/Bishkek
Europe/Rome	Europe/Tiraspol	Asia/Dushanbe
Europe/San_Marino	Europe/Uzhgorod	Asia/Karachi
Europe/Sarajevo	Europe/Vilnius	Asia/Oral
Europe/Skopje	Europe/Zaporozhye	Asia/Samarkand
Europe/Stockholm	Israel	Asia/Tashkent
Europe/Tirane	Libya	Asia/Yekaterinburg
Europe/Vaduz	Turkey	Etc/GMT-5
Europe/Vatican	Africa/Addis_Ababa	Indian/Kerguelen
Europe/Vienna	Africa/Asmera	Indian/Maldives
Europe/Warsaw	Africa/Dar_es_Salaam	PLT
Europe/Zagreb	Africa/Djibouti	Asia/Calcutta
Europe/Zurich	Africa/Kampala	IST
MET	Africa/Khartoum	Asia/Katmandu
Poland	Africa/Mogadishu	Antarctica/Mawson
ART	Africa/Nairobi	Antarctica/Vostok
Africa/Blantyre	Antarctica/Syowa	Asia/Almaty
Africa/Bujumbura	Asia/Aden	Asia/Colombo
Africa/Cairo	Asia/Baghdad	Asia/Dacca
Africa/Gaborone	Asia/Bahrain	Asia/Dhaka
Africa/Harare	Asia/Kuwait	Asia/Novosibirsk
Africa/Johannesburg	Asia/Qatar	Asia/Omsk
Africa/Kigali	Asia/Riyadh	Asia/Qyzylorda
Africa/Lubumbashi	Asia/Tbilisi	Asia/Thimbu
Africa/Lusaka	EAT	Asia/Thimphu
Africa/Maputo	Etc/GMT-3	BST
Africa/Maseru	Europe/Moscow	Etc/GMT-6
Africa/Mbabane	Indian/Antananarivo	Indian/Chagos
Africa/Tripoli	Indian/Comoro	Asia/Rangoon
Asia/Amman	Indian/Mayotte	Indian/Cocos
Asia/Beirut	W-SU	Antarctica/Davis
Asia/Damascus	Asia/Riyadh87	Asia/Bangkok
Asia/Gaza	Asia/Riyadh88	Asia/Hovd
Asia/Istanbul	Asia/Riyadh89	Asia/Jakarta
Asia/Jerusalem	Mideast/Riyadh87	Asia/Krasnoyarsk
Asia/Nicosia	Mideast/Riyadh88	Asia/Phnom_Penh
Asia/Tel_Aviv	Mideast/Riyadh89	Asia/Pontianak
CAT	Asia/Tehran	Asia/Saigon
EET	Iran	Asia/Vientiane
Egypt	Asia/Baku	Etc/GMT-7
Etc/GMT-2	Asia/Dubai	Indian/Christmas
Europe/Athens	Asia/Muscat	VST
Europe/Bucharest	Asia/Yerevan	Antarctica/Casey
Europe/Chisinau	Etc/GMT-4	Asia/Brunei

Asia/Chongqing	Australia/Lindeman
Asia/Chungking	Australia/Melbourne
Asia/Harbin	Australia/NSW
Asia/Hong_Kong	Australia/Queensland
Asia/Irkutsk	Australia/Sydney
Asia/Kashgar	Australia/Tasmania
Asia/Kuala_Lumpur	Australia/Victoria
Asia/Kuching	Etc/GMT-10
Asia/Macao	Pacific/Guam
Asia/Macau	Pacific/Port_Moresby
Asia/Makassar	Pacific/Saipan
Asia/Manila	Pacific/Truk
Asia/Shanghai	Pacific/Yap
Asia/Singapore	Australia/LHI
Asia/Taipei	Australia/Lord_Howe
Asia/Ujung_Pandang	Asia/Magadan
Asia/Ulaanbaatar	Etc/GMT-11
Asia/Ulan_Bator	Pacific/Efate
Asia/Urumqi	Pacific/Guadalcanal
Australia/Perth	Pacific/Kosrae
Australia/West	Pacific/Noumea
CTT	Pacific/Ponape
Etc/GMT-8	SST
Hongkong	Pacific/Norfolk
PRC	Antarctica/McMurdo
Singapore	Antarctica/South_Pole
Asia/Choibalsan	Asia/Anadyr
Asia/Dili	Asia/Kamchatka
Asia/Jayapura	Etc/GMT-12
Asia/Pyongyang	Kwajalein
Asia/Seoul	NST
Asia/Tokyo	NZ
Asia/Yakutsk	Pacific/Auckland
Etc/GMT-9	Pacific/Fiji
JST	Pacific/Funafuti
Japan	Pacific/Kwajalein
Pacific/Palau	Pacific/Majuro
ROK	Pacific/Nauru
ACT	Pacific/Tarawa
Australia/Adelaide	Pacific/Wake
Australia/Broken_Hill	Pacific/Wallis
Australia/Darwin	NZ-CHAT
Australia/North	Pacific/Chatham
Australia/South	Etc/GMT-13
Australia/Yancowinna	Pacific/Enderbury
AET	Pacific/Tongatapu
Antarctica/DumontDURville	Etc/GMT-14
Asia/Sakhalin	Pacific/Kiritimati
Asia/Vladivostok	
Australia/ACT	
Australia/Brisbane	
Australia/Canberra	
Australia/Currie	
Australia/Hobart	