

SenseLab

Open sensors monitoring and device control in Your lab

Overview

SenseLab is planned to be a sensor data acquisition, visualization and device control software system with open and extensible architecture to be used in Your lab. Nodes of devices or sensors are processed to create meaningful measurements data logging allowing to replay recorded measurement sessions, visualize and analyze it also with the help of external tools. Basically this is [SCADA](#) like system.

Architecture

Actions done in a lab or any other place are performed at some environment where interactions occur during the progression of events we call time. The environment is monitored or controlled by nodes.

Nodes

Nodes provide the means of communication with measured and controlled space or scene. Environment nodes represent organized devices and project nodes represent an interpretation of environment nodes interactions during actions in a project.

Environment nodes

Environment nodes are organized devices with sensors or controllers which provide or process data and are part of node network being monitored or controlled. Each node has

- Identifier
- Name
- Description
- Recordables
- Optional subnodes
 - The list can be fixed or dynamic (when nodes appear or disappear)
 - Nodes can be provided by monitoring networks of nodes which can be discovered

Example

- Bluetooth Low Energy device like
 - [Texas Instruments SensorTag](#)
 - [Mooshim Engineering Mooshimeter](#)

- Device part like
 - Magnetometer
 - Thermometer
 - Motor speed control

Node recordables

Nodes provide recordables or recordable items which allow recording of records (changes or actions done in a node). Each record has

- Recorded data
- Spatial location
- Temporal location

These are types of recordables:

- Values
 - Represent read only or writable states or measurements
 - Each value has
 - Measured or written value
 - Optional value unit (eg. K - Kelvin)
 - Optional value quantity (eg. Temperature)
- Commands
 - Represent actions initiated by a user and performed on a node
 - Can have specific parameters for their control

Project nodes

Project nodes represent environment nodes in a project. Each node has

- Identifier
- Name
- Description
- Environment node used
- Whether a node is selected
- Selected recordables from an environment node
- Spatial location
- Optional subnodes

Node locations

Each node has its spatial location which can be of different kinds like:

- Name
- Geolocation like GPS
- Absolute coordinates like x, y, z
- Coordinates relative to parent node
- Group of the previous ones

Projects

Project represents a set of project nodes and sequences of records.

Records

Selected recordables of selected project nodes can be recorded into a named sequence of records. Records provided by environmental recordables (while real actions are performed on stage during a period of time) can be

- Stored directly
- Transformed to another new record(s) and stored

Records can be played during recording or replayed later once recorded. Records can be visualized directly in this software or indirectly by some external visualization software where we can feed data during the records playback and analyse it later. Such software can be something like

- [Visit](#)
- [MayaVi](#)

Example

Experiments with dynamic plasmatic reactors may require magnetic field mapping, reactor control etc.

Let us suppose such control can be done with reactor vacuum chamber controllers dealing with

- Vacuum level
- Gas injection
- Chamber rotation speed

We could add more, but let us keep it simple for now.

From this we get these environment nodes

- Vacuum chamber pressure sensor
- Vacuum chamber valves controller
- Vacuum pump controller
- Motor controller with feedback for obtaining rotational speed, voltage and current

Regarding magnetic field mapping, we can use multiple magnetometers as environment nodes.

The whole environment tree would be:

- My lab
 - Reactor
 - Vacuum chamber
 - Chamber pressure
 - Vacuum valve

- Gas injection valve
 - Motor
 - RPM
- Vacuum pump
 - Switch
- Magnetometers
 - M1
 - M2
 - M3
 - M4

We will represent these environment nodes of our lab as project nodes in our experimental project. We will perform actions leading to plasma opening in our reactor. During these actions we will record states of our controllers and vacuum chamber sensors and at the same time magnetic field outside of the chamber.

The whole project tree would be:

- My lab
 - Reactor
 - Vacuum chamber
 - Chamber pressure
 - Vacuum valve
 - Gas injection valve
 - Motor
 - RPM
 - Vacuum pump
 - Switch
 - Magnetic field
 - M1
 - M2
 - M3
 - M4

The magnetic field recordings consisting of magnetometer B (bi)vector field readings at specific points can be visualized.

Node storage

Environment and project nodes can be stored in different types of storage like

- Local
- Remote
- XML
- Database

These storages would allow also remote access to environments and projects by other users so that they can learn, share, contribute for better cooperation.

Technology

From technological point of view, I suggest these technologies:

System requirements

For system running

- [Windows 8.1](#)

For system development

- [Windows 8.1](#)
 - Provides [Bluetooth Low Energy API](#) ¹⁾ as Windows Runtime component
- [.NET Framework 4.5](#)
 - Programming platform and runtime environment
- [WPF](#) ²⁾
 - For UI ³⁾
- [Prism](#)
 - Composite extensible application framework
- [Visual Studio Express 2013 for Windows Desktop](#)
 - Programming environment
 - Git repository access

Application structure

Application functionality can be divided into these blocks

- Main application with
 - MVVM ⁴⁾ pattern
 - Projects editor, monitor and player
- WCF ⁵⁾ service with
 - Node storage access
- Database for data storage like
 - SQLite
 - Cloud storage

Cooperation

Anybody is welcome to be part of development team or just make suggestions or share ideas.

Project repository

Project code repository can be accessed at <https://github.com/OMLN/SenseLab>

- ¹⁾ Application Programming Interface
- ²⁾ Windows Presentation Foundation
- ³⁾ User Interface
- ⁴⁾ Model, view model, view
- ⁵⁾ Windows Communication Foundation

From:

<https://openmagneticlight.net/> - **Open Magnetic Light Network**

Permanent link:

<https://openmagneticlight.net/projects/software/senselab>

Last update: **2014/09/21 21:36**

