EIT ICT Labs

Call for SME for Activity 11831 for 2014

A software API for a vendor agnostic framework for wireless sensor network gateways for Smart Energy Systems

Context and goal

The EIT Activity “Open SES Experience Labs” within the action line “Smart Energy Systems” aims to develop technologies for sensing, modeling and visualization of energy to provide tools (“Apps”) that are easy to use and adapted to the need of ordinary consumers.

Wireless sensor networks (WSN) are key technologies in this context, and many of them, such as Zigbee, EnOcean, Z-wave are now widely available on the market. Meanwhile, semiconductor manufacturers offer powerful low-consumption CPUs and motherboards that allow integrators to come up with gateways, based on proprietary software, and specialized for a particular application domain such as home automation, smart metering, safety of people and property, or even healthcare.

Each gateway embeds a set of functions as a monolithic firmware that cannot be easily updated or augmented by the final user, and none of these gateways can interoperate. These facts lead to an unnecessary market fragmentation that is a significant lock for a massive spread of systems based on WSN. Therefore, it is important for this industry to have a vendor agnostic software framework to ensure compliance between gateways and third party software, just like the Android™ operating system ensure to mobile application developers a huge range of compliant target devices.

If Android™ can be seen as a vendor agnostic framework suitable for any devices with a display, WSN gateways mostly do not have one, and provide web based user interface (UI) instead. Then, the WSN industry lacks an Android™ equivalent framework suitable for WSN gateways. The goal of this call is to provide an open specification of such a framework to the community. This specification will exist as a software API that can be freely implemented by any gateway manufacturers. This software API will be freely available for third party developers who wish to offer applications for a large range of WSN gateways.

As part of this call, the SME partner shall provide a specification of the expected API and a demonstrating implementation of that API that can be used by activity partners. This demonstrating implementation must run on a WSN gateway available on the market. Also, the SME partner shall cooperate with one activity’s partner to produce a first application on the theme «Make energy visible».

Next chapter will present general requirements for the expected API, then chapter 3 focuses on requirements about the expected API itself. Finally, chapter 4 presents requirements about the demonstrating application.

General requirements

This chapter presents requirements in terms of a general architecture of the expected framework, and then in terms of targeted software technologies.

Architecture

The expected vendor agnostic framework aims to ease the development and hosting of software applications that need to deal with sensors and actuators provided by various WSN technologies. Applications can be dynamically installed / uninstalled in/from the system. By dynamically, we mean during system execution, without need to restart the system.

In this context, an application is a software entity that (see figure 1)
- embeds some logic to process data coming from sensors and actuators,
- could exchange data over the network with third party systems,
- could interact with users through a web based user interface (Web UI),
- and relies on the framework for extra-functional services such as database, web server, or publish/subscribe messaging systems.

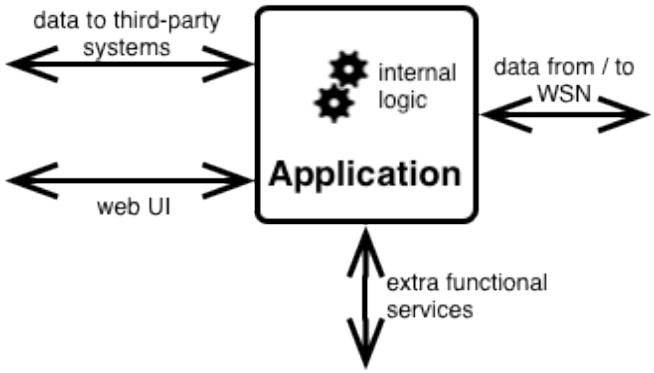

Figure 1. An application in the context of a WSN gateway.

To ensure development easiness, application developers do not have to know the details of the underlying WSN technologies, and applications have to be as independent as possible from the underlying WSN technologies. Therefore, interactions between applications and WSN need to be done through an appropriate abstraction (see figure 2).

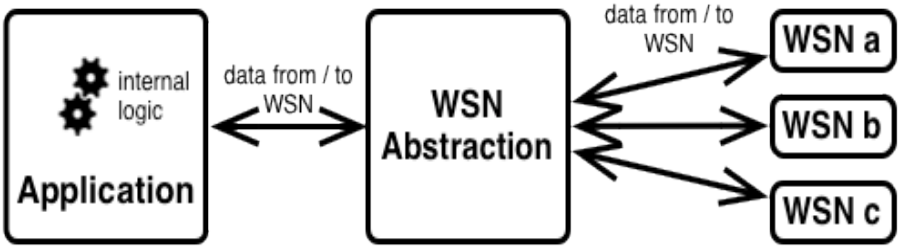

Figure 2. The WSN abstraction concept.

However, a single abstraction of a WSN will probably never fit all contexts of use and all existing and future WSN technologies. Then, rather than a single abstraction, the expected vendor agnostic framework must include a mediation layer.

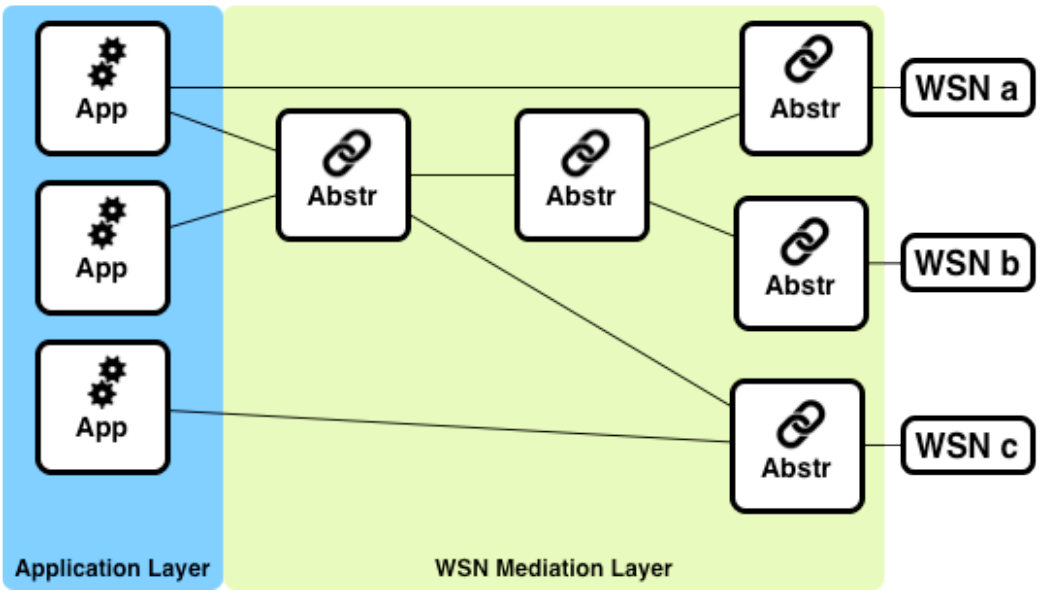

Figure 3. The expected WSN mediation layer.

The purpose of this mediation layer is to host a set of WSN abstractions and to organize them into mediation chains. Each chain link transforms data from a lower level of abstraction to a higher level of abstraction. Applications should be able to connect at any levels of any mediation chains to get the most appropriate ones (see figure 3).

Like applications, WSN abstractions can be installed / uninstalled in/from the system dynamically. Also, like applications, WSN abstractions can be developed and provided by third parties.

Technologies
It is expected that the framework is implemented in Java Standard Edition, JDK 1.7 and must rely on the OSGi Release 4.3 framework and the iPOJO v1.11.0 service component model.

Therefore, it is expected that the delivered framework API is packaged as an OSGi bundle by using the standard functionalities of OSGi R4.x core framework specifications.

The framework API has to be delivered along with a reference implementation, a reference SDK and its documentation. However, the source code of the reference implementation and of the SDK do not need to be public.

API requirements

The software API must cover all expected functions of the vendor agnostic framework. These functions are organized around 3 topics: application, WSN abstraction and extra-functional services.

Application

- Management of application life cycle: the API must provide functions to install, uninstall, start, stop and update an application.

- Management of interaction with WSN abstractions: the API must provide functions to let applications discover and connect to any relevant WSN abstractions that run in the system.

- Web UI: the expected specifications must normalize a common structure to organize web resources of application web UI. Hence, each application will proceed the same way to provide its icon, access to its configuration UI.

WSN Abstraction

- Management of WSN abstraction life cycle: the API must provide functions to install, uninstall and update a WSN abstraction.

- Management of interaction with other WSN abstractions: the API must provide functions to let WSN abstractions discover and connect to any relevant other WSN abstractions that run in the system.

- Web UI: the expected specifications must normalize a common structure to organize web resources of WSN abstraction web UI. Hence, each WSN abstraction will proceed the same way to provide its UI to users.

Additional functional services

- Database for local storage: the API must provide functions to let applications and WSN abstractions get a connection to a local database system.

- Access to a web server: the API must provide functions to let applications and WSN abstractions expose services or resources through a web server.

- Event broadcasting: the API must provide functions to let applications and WSN abstractions publish and subscribe to events.

These requirements only make up a minimal set of additional functional services. Any relevant addition can be proposed by the SME partner.

Application requirements

The expected application will be the main collaboration opportunity between the SME partner and the activity partner. Thematically this application will be about «Make Energy Visible».
This application should target the rising market of energy saving at home. It aims to collect data through meters about the energy consumption at home/office and then showing the user how much energy is consumed and how to set up strategies to save it.

General conditions:

- The funding is 40.000€, which includes personal costs and traveling costs
- The SME will join the EIT ICT Activity from 01.01.2014 - 31.12.2014
- The call will close at 18th of December, decision about participation is January 7th

Proposals shall be submitted in written from and English to: Dagmar Koss at koss@fortiss.org