



Project FoSIBLE
Fostering Social Interactions for a Better Life of the Elderly

***D3.4 – Report on proof of concepts according integration feasibility
with platform***



Responsible

Fraunhofer IMS

Mauser Care

Deliverable

D3.4 – Report on proof of concepts according integration feasibility with platform

Version: 1.0

Date: 31/01/13

Dissemination level: (PU, PP, RE, CO): PU

Abstract

The objectives of this work package are to develop and build prototypes of the interfacing devices that can be used to prove the new interaction concepts. These prototypes will be a combination of sensors, actuators and furniture. As a result, the goal of this work package is to develop “smart” furniture that accompanies TV based applications by natural and intuitive user input and output options. This is the third deliverable from work package 3 and the scope is to describe the interaction between the input and output devices and the application platform.

Table of Content

- 1. Introduction 4
 - 1.1 Related Tasks 4
 - 1.2 Scope of this Deliverable..... 4
- 2. Basic connection techniques 5
 - 2.1 enOcean 5
 - 2.2 Radio-frequency identification 6
 - 2.3 HTTP communication..... 7
 - 2.4 Ethernet 10
- 3. Integration of input and output devices into the social community platform..... 11
 - 3.1 Invitation to take a walk 11
 - 3.2 Invitation for sport..... 11
 - 3.3 Invitation for Nordic Walking..... 13
 - 3.4 Invitation for coffee 13
 - 3.5 Invitation for board games 13
 - 3.6 Post your book 13
 - 3.7 Notification lamp and motion detector..... 14
 - 3.8 TV show reminder..... 16
 - 3.9 Deactivate notifications..... 16
 - 3.10 Configuration of activity rules..... 16
 - 3.11 System wide shut down 16
- 4. References 18

1. Introduction

1.1 Related Tasks

Task 3.1: (Responsible: Mauser Care, Collaboration: FhG IMS) Development of concepts for innovative devices/furniture, with sensors for input and output. This includes creating concepts on how to integrate sensor data from the environment into the social media center applications. Current plans target at sitting furniture, beds as well as wardrobes and sideboards. Note that in order to come up with prototypes within a short time frame, only sensors that are available on the market will be considered. Currently, application of the following sensor types is considered: proximity sensors (capacitive measurement), RFID-based sensors, infrared based sensors. Note that this list is not exhaustive.

Task 3.2: (Responsible: Mauser Care, Collaboration: FhG IMS) Development of proof of concepts (prototype based) on the basis of the developed concepts for testing validity and acceptance by end users. Development will be done in two phases. During the first phase, mock-ups will be created that provide the test users with look and feel of the final devices. These mock-ups will be used to gather first user feedback. In the next phase, a set of full functional prototypes will be created that will then be used to gain detailed and exhaustive end user feedback.

Task 3.3: (Responsible: Mauser Care, Collaboration: FhG IMS) Integration of hardware and software components (platform). These prototypes will make use of standard communication solutions (wireless) to connect the sensors to the application platform.

Task 3.4: (Responsible: Mauser Care, Collaboration FhG IMS) Combine the devices for input, output and the software applications in order to make them usable for remote interaction.

Task 3.5: (Responsible: FhG IMS, Collaboration Mauser Care) Development of hardware interfaces for various spatial scenarios (e.g. different rooms). This task also addresses using common off the shelf domotic sensors (e.g., motion detectors) in order to gather valuable information about the current location of the habitant.

Task 3.6: (Responsible: AIT, Collaboration Mauser Care) Integration of a vision-based sensor technology into a furniture object to explore the posture and facial expression of the users.

1.2 Scope of this Deliverable

The objectives of this work package are to develop and build prototypes of the interfacing devices that can be used to prove the new interaction concepts. These prototypes will be a combination of sensors, actuators and furniture. As a result, the goal of this work package is to develop “smart” furniture that accompanies TV based applications by natural and intuitive user input and output options. This is the third deliverable from work package 3 and the scope is to describe the interaction between the input and output devices and the application platform.

2. Basic connection techniques

In this chapter all basic communication and technologies are described which are used during the interaction processes between the devices and the application platform. Figure 3 illustrates the general communication of the social community platform, which is possible via Ethernet.

2.1 enOcean

As mentioned in Deliverable 3.1 the “Mauser Runtime” consists of associated device controllers that are responsible for processing the device data. The devices are kept abstract, so that a replacement of a device does not disrupt the system itself. The main device controller is the “enOcean” controller. It is responsible for the integration of multiple sensors of the correspondent building automation bus.

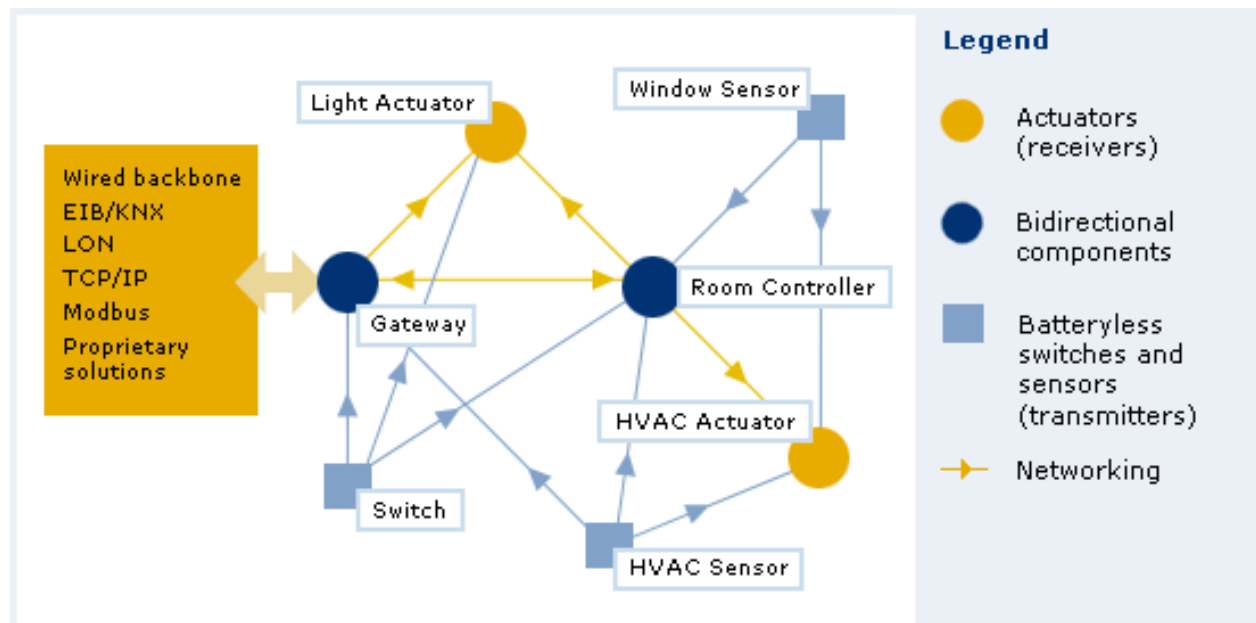


Figure 1: An energy harvesting wireless network from "enOcean" [1]

Most of “enOcean” components use the principle of “Energy Harvesting”. This principle is based on basic physical observations: If state parameters are changed somewhere, an energy form is altered into another form of energy. For example if a sensor is moved, the kinetic energy is sufficient enough to create another form of energy in order to transmit digital information from one source to another. The “enOcean” components are powered by this energy and transmit information wireless, by the components fulfill the ISO/IEC 14543-3-10 standard for ultra-low power consumption. The individual components are not able to communicate with other components directly. Each component has to communicate with one or more so called gateways. These gateways fulfill this task and transfer the information

to other communication partners. If the “enOcean” component is an actor, the communication has to be bidirectional. An example of wireless network can be seen in the Figure 1.

2.2 Radio-frequency identification

In addition to the “enOcean” controller a Radio-frequency identification (RFID) controller has been added to the “Mauser Runtime”. RFID is a wireless non-contact system that uses radio-frequency electromagnetic fields to transfer data from a tag attached to an object to a sensor which can read RFID tags. The tag does not need to be within the line of sight of the reader and may be embedded in the object. Purpose of this technique is the identification and tracking of objects. Figure 2 shows several RFID tags in different sizes which can be attached to objects. Shape and size of the tags influence the transmission range.



Figure 2: RFID tags in different sizes [2]

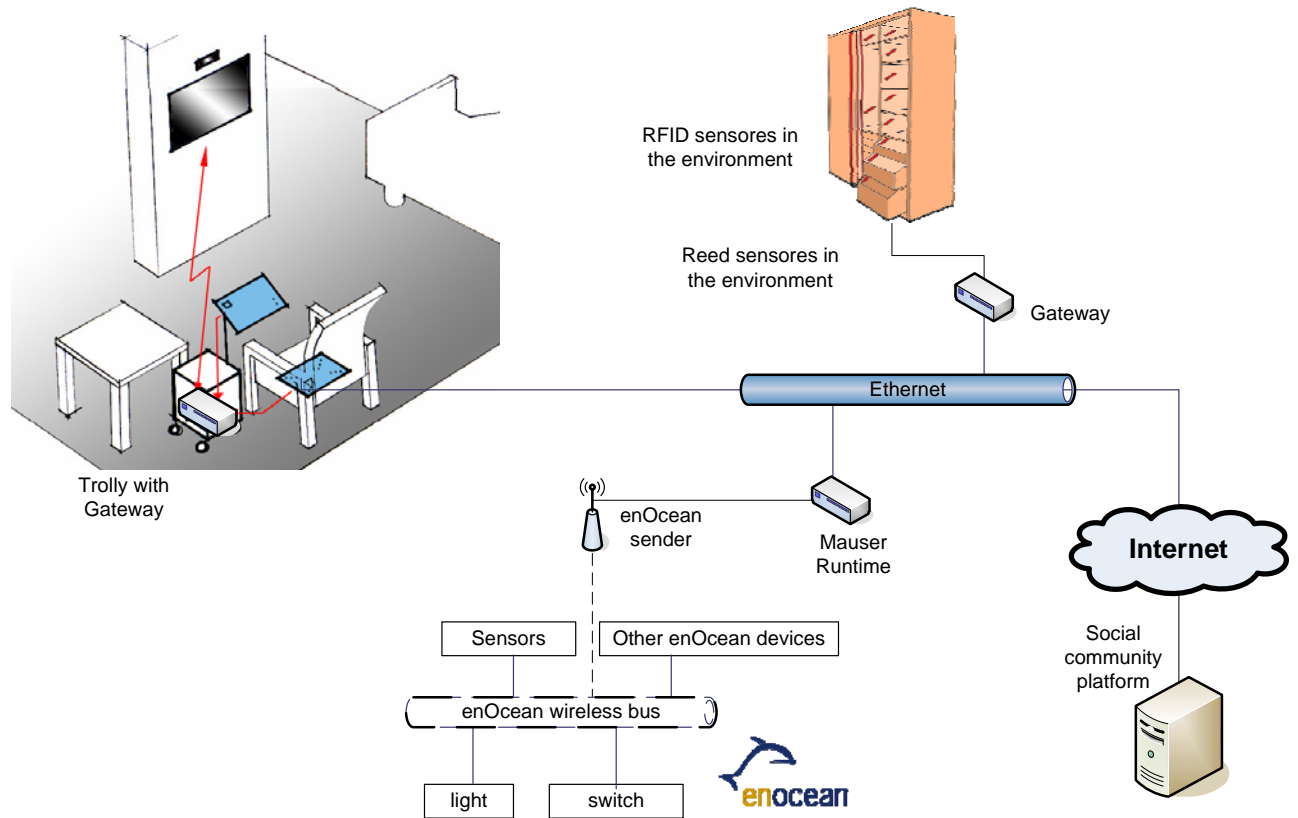


Figure 3: Communication between the components

2.3 HTTP communication

In computing, POST is one of many request methods supported by the HTTP protocol used by the World Wide Web. The POST request method is designed to request that a web server accepts the data enclosed in the request message's body for storage. For example it can be used to upload a file. The POST request communication is used by the communication with the social community platform. In this communication, Extensible Markup Language (XML) files are exchanged, which include commands for the remote communication partner. Figure 4 shows an XML file that includes a method called "insert". There are also four values which are used as parameters for that method.

```

<?xml version="1.0" ?>
- <methodCall>
  <methodName>insert</methodName>
- <params>
  - <param>
    - <value>
      <int>63</int>
    </value>
  </param>
  - <param>
    - <value>
      <string>Sensor 1</string>
    </value>
  </param>
  - <param>
    - <value>
      <double>36.0</double>
    </value>
  </param>
  - <param>
    - <value>
      <string>params</string>
    </value>
  </param>
</params>
</methodCall>

```

Figure 4: HTTP Post FoSIBLE communication, XML Scheme

The communication between the “Mauser Runtime” and the smart TV application also uses the HTTP protocol. If the client (“Mauser Runtime” and tablet PC application) wants to send messages to the smart TV application, it also uses HTTP Post. Therefore the “Mauser Runtime” and the tablet PC application have to register as clients at the smart TV before they can exchange information with the TV applications. Once this is done, a client can send HTTP Post requests whenever it has messages for the TV application (e.g. the cane has been drawn from its rack, see chapter 3.1 for more details). Figure 5 shows the communication process in each step.

If the smart TV application wants to send messages to the registered clients (“Mauser Runtime” and tablet PC application), a long polling HTTP Get is used. Again the clients have to first register before any messages can be exchanged. Basically the client sends a GET request

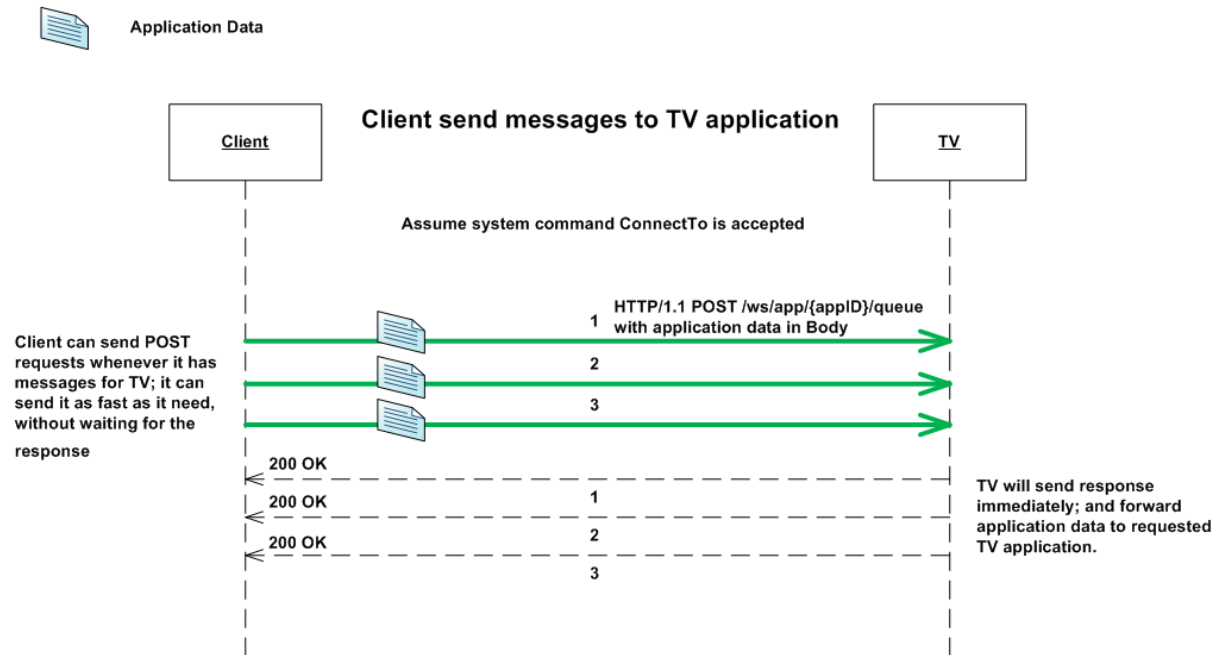


Figure 5: HTTP Post communication between Mauser Runtime (client) and the smart TV application [3]

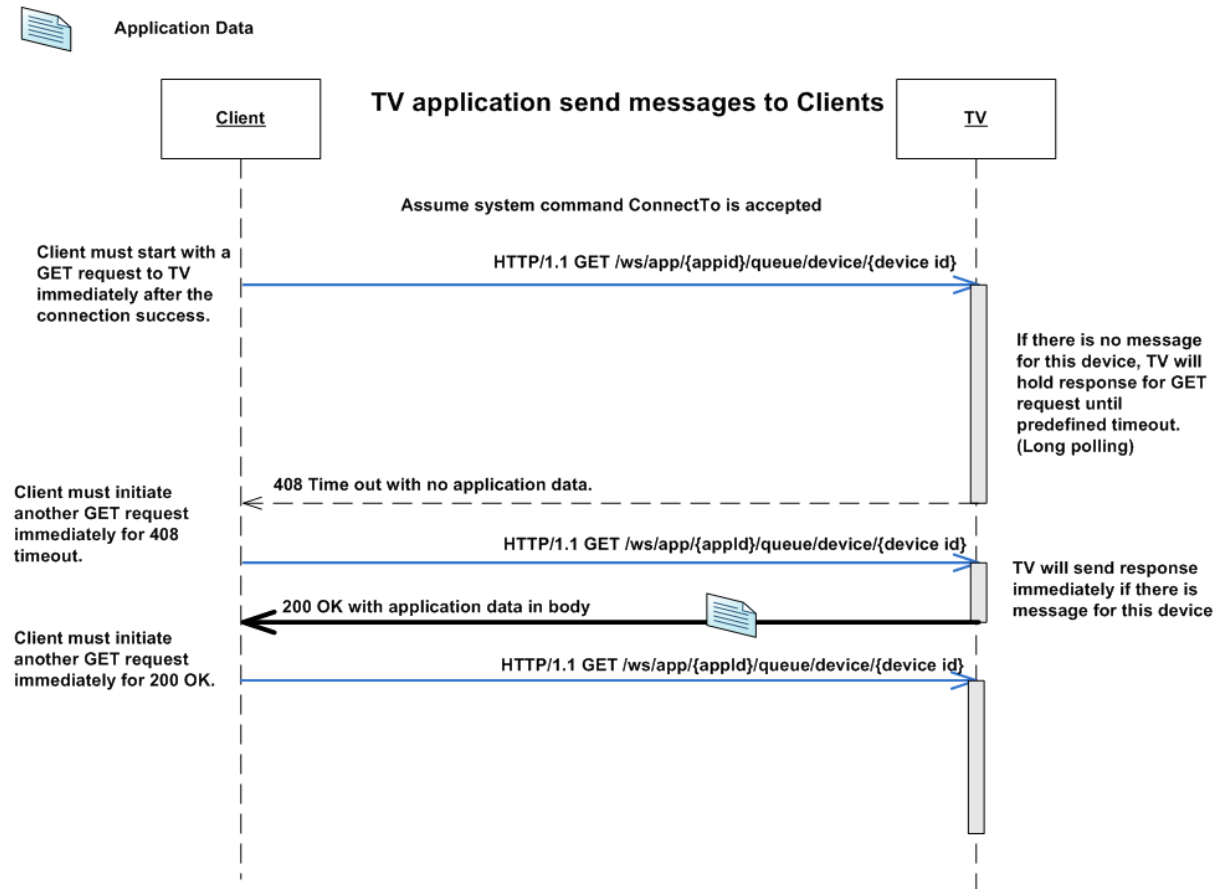


Figure 6: Long polling HTTP Get communication between the smart TV application and the Mauser Runtime (client) [3]

and if there's a message for this device, the smart TV application sends the data immediately as response back to the client. Otherwise the TV application waits until a message for this device is available and then sends it as a response back to the client. Should a predefined timeout occur while waiting or if the TV application sends a message to the client, the whole process is repeated. Figure 6 shows the communication process in each step.

2.4 Ethernet

For connecting external applications to the “Mauser Runtime” it includes an interface that can be addressed via Ethernet/TCP.

3. Integration of input and output devices into the social community platform

In this chapter the ongoing integration of the developed smart input and output devices into environment and their use in the social community platform is described. They utilize all previously shown communication techniques in chapter 2. The focus of this chapter lies on the description of the concepts for integration the input and output devices into the social community platform.

3.1 Invitation to take a walk

Several objects in the environment can be used as input devices for interaction with the application platform. One of these objects is a cane enhanced with a small RFID chip (see additional information in chapter 2). As shown in Figure 7, the cane is placed in a corresponding cane rack, where an ambient RFID reader can detect the tag. Upon drawing the cane from a cane rack the RFID reader sends this information to the “Mauser Runtime” via Ethernet. If the tag does not belong to the cane (there are several tags in the environment), the information is ignored. Otherwise the “Mauser Runtime” forwards this information again via Ethernet to the smart TV application (HTTP Post). Depending on the application settings the user has the possibility to invite his social community friends for a walk. If the user agrees to invite a friend, the invitation is send from the smart TV application via Internet. The communication process is handled by a HTTP request, where the invitation is encapsulated as an XML file.

3.2 Invitation for sport

Similar to the cane described in chapter 3.1, a sports bag is outfitted with a small RFID chip. The sports bag is placed in a basket, where a hidden RFID reader can read the tag. Upon removing the sports bag from the basket the RFID reader sends this information to the “Mauser Runtime” via Ethernet. If the tag does not belong to the sports bag (there are several tags in the environment), the information is ignored. Otherwise the “Mauser Runtime” forwards this information again via Ethernet to the smart TV application (HTTP Post). Depending on the application settings the user has the possibility to invite his social community friends for sports. If the user agrees to invite a friend, the invitation is send from the smart TV application via Internet. The communication process is handled by a HTTP request, where the invitation is encapsulated as an XML file.

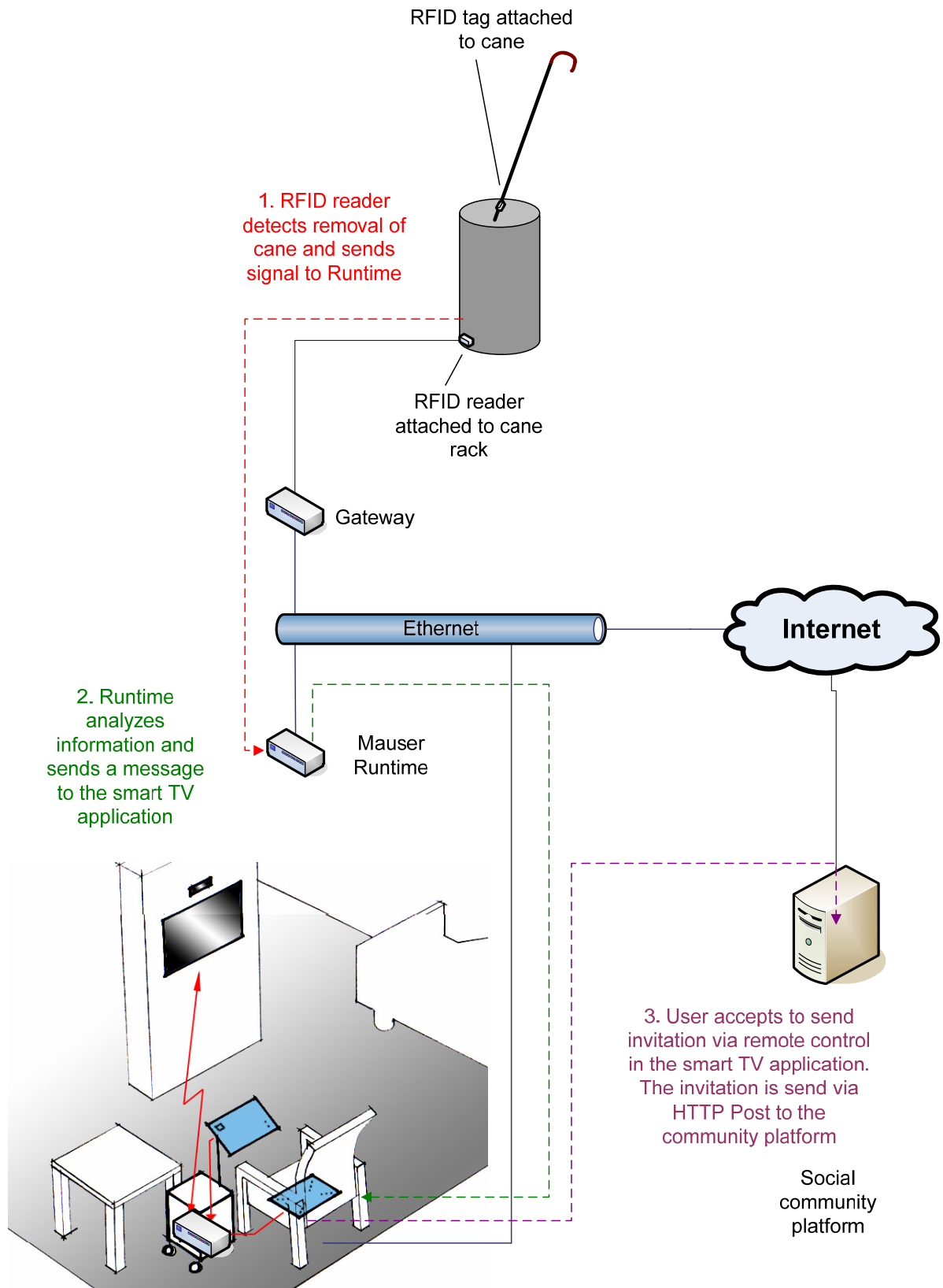


Figure 7: General flow of events during the invitation process

3.3 Invitation for Nordic Walking

Similar to chapter 3.1, a Nordic Walking pole is enhanced with a small RFID chip. The Nordic Walking pole is placed in a rack, where a ambient RFID reader can detect the tag. Upon removing the pole from the rack the RFID reader sends this information to the “Mauser Runtime” via Ethernet. If the tag does not belong to the Nordic Walking pole (there are several tags in the environment), the information is ignored. Otherwise the “Mauser Runtime” forwards this information again via Ethernet to the smart TV application (HTTP Post). Depending on the application settings the user has the opportunity to send an invitation to his social community friends for Nordic Walking. If the user agrees to invite a friend, the invitation is sent from the smart TV application via Internet. The communication process is handled by a HTTP request, where the invitation is encapsulated as an XML file.

3.4 Invitation for coffee

Similar to chapter 3.1, several pieces of a coffee service are enhanced with small RFID chips. The coffee service is placed in a cupboard, where a hidden RFID reader can detect the tags. Upon removing pieces of the coffee service from the cupboard the RFID reader sends this information to the “Mauser Runtime” via Ethernet. If the tags don’t belong to the coffee service (there are several tags in the environment), the information is ignored. Otherwise the “Mauser Runtime” forwards this information again via Ethernet to the smart TV application (HTTP Post). Depending on the application settings the user has the opportunity to send an invitation to his social community friends for coffee and cake. If the user agrees to invite a friend, the invitation is send from the smart TV application via Internet. The communication process is handled by a HTTP request, where the invitation is encapsulated as an XML file.

3.5 Invitation for board games

As described in chapter 3.1, each board game box is enhanced with small RFID chips. The boxes are placed in a closet, where a hidden RFID reader can detect the tags. Upon removing one ore more games of the closet the RFID reader sends this information to the “Mauser Runtime” via Ethernet. If the tags don’t belong to any games (there are several tags in the environment), the information is ignored. Otherwise the “Mauser Runtime” forwards this information again via Ethernet to the smart TV application (HTTP Post). Depending on the application settings the user has the opportunity to invite his social community friends for some hours of board gaming. If the user agrees to invite a friend, the invitation is sent from the smart TV application via Internet. The communication process is handled by a HTTP request, where the invitation is encapsulated as an XML file.

3.6 Post your book

As mentioned in chapter 2 RFID tags can be attached to objects in order to identify these with a RFID reader. Using several unique RFID tags different books can be distinguished from each other. Every book is equipped with one tag (spine or cover of the book) while the

reader is installed in a special area beyond the surface of a coffee table. If the user puts a book in the specific area of the coffee table, the reader recognizes the tag even through the material of the coffee table and sends the tag information to the “Mauser Runtime”. The RFID controller handles the data and the Runtime transmits it via Ethernet to the smart TV application (HTTP Post). Here the tag information is compared with book entries in a data base. As soon the book is identified and depending on the application settings the user now has the opportunity to share the book information with his social community friends.

If the user agrees to share the book information with one or more friends, the data is sent from the smart TV application via Internet to the social community platform. The communication process is handled by a HTTP post request, where the invitation is encapsulated as an XML file in the POST request.

3.7 Notification lamp and motion detector

The message lamp is a standard lamp which can be controlled by the “enOcean” automation bus. If the user receives a message from a friend in the social community platform or if he set a reminder in the smart TV application – short, if he gets a notification, then the lamp is switched on. It serves as a visual information feedback.

There are two sources of notifications: Local and social community platform notifications. If the source of the notification is a component of the social community platform, the information is send via HTTP Post to the smart TV application. Local notifications are directly generated by the smart TV application. Depending on the application settings the notification will be dismissed or it will be send via Ethernet to the “Mauser Runtime” (HTTP Post). Here the data is analyzed and interpreted as “notification activation”. The “enOcean” controller triggers the notification lamp via “enOcean” automation bus. As soon as the user reacts to the notification presented by in the smart TV application, the TV sends the appropriate event to the “Mauser Runtime” in order to switch off the lamp.

For full availability of the user, a notification lamp should be installed in each room of the apartment or house. Since the user can only be in one room at a time, switching on all lamps would be redundant. Therefore each lamp is combined with a motion detector. So before a lamp is switched on, the “Mauser Runtime” verifies the state of movement via “enOcean” motion detectors. Only in rooms where motion is detected, the notification lamps are turned an.

Figure 8 illustrates the whole communication process between the social communication platform, the “Mauser Runtime” and the output device (notification lamp) during a notification.

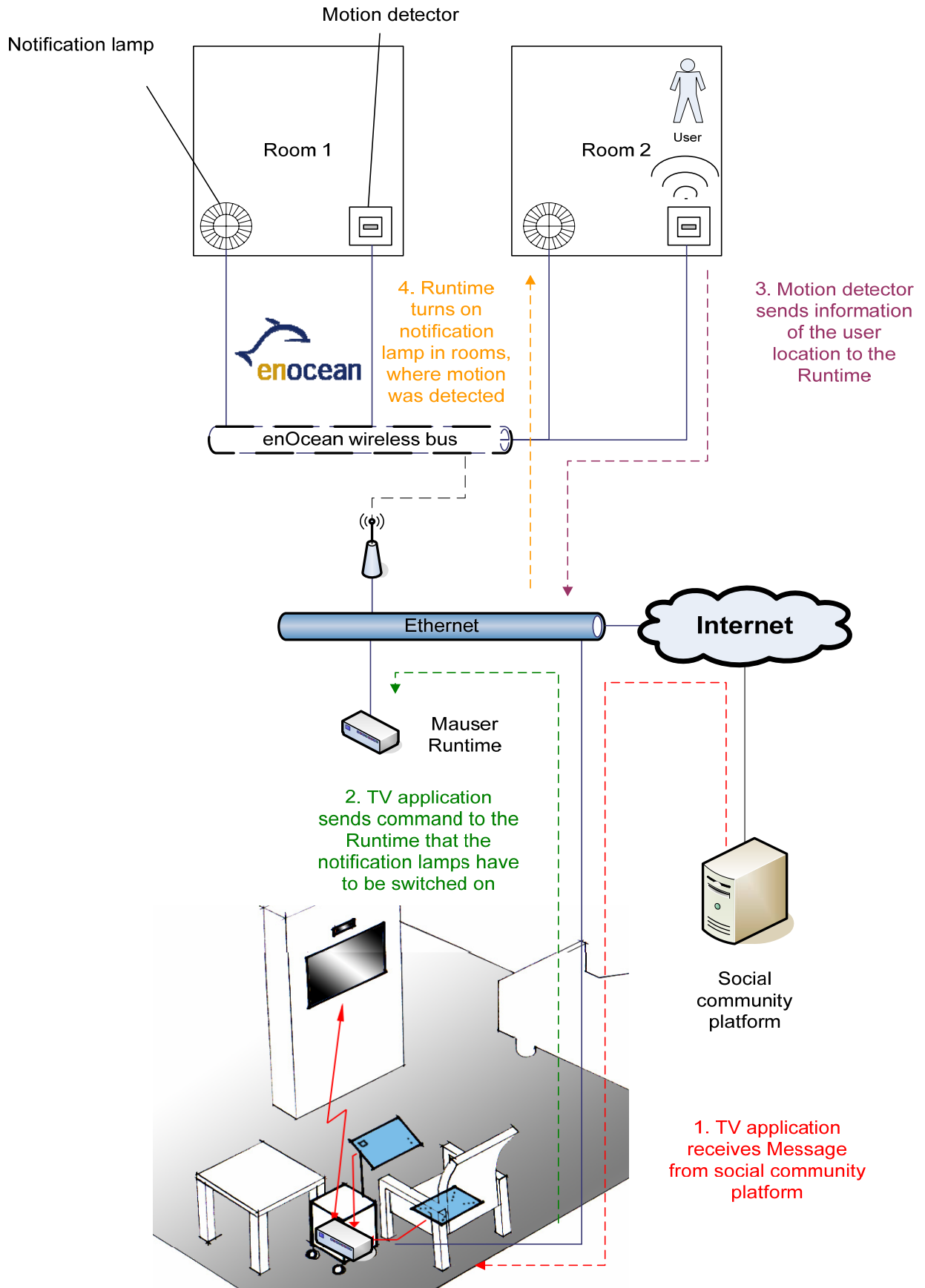


Figure 8: General flow of events during an notification

3.8 TV show reminder

The user has the option to activate a reminder function for TV shows. Several minutes before the show starts, the user gets a notification pop up message in the smart TV application. The reminder message is shown on the screen until the user confirms the receipt of the message via remote control. In addition to the notification in the smart TV application the environment notifies the user by means of a notification lamp. See chapter 3.7 for more information.

3.9 Deactivate notifications

The user has the option to deactivate all notifications – including smart TV notifications and the notification lamp. The local notifications are directly discarded, because they are generated by the smart TV application. Notifications from the social community platform are sent to the smart TV application by default. Once the notification message reaches the smart TV application, it is discarded like the local notification message.

3.10 Configuration of activity rules

In the range of this project, we need to gather information from the environment in order to analyze which activity the user is currently executing. We obtain the information by different sensors in the environment, e.g. Reed-contacts in drawers or motion detectors. To define what a walk is or to describe in general how certain possible activities in the environment are performed, we developed a “rule configurator” application for the portable tablet. As soon the application is started, it scans available sensors in the environment and pre-defined activities are listed and individually selectable. After selecting one or more sensors, the user must set a name for that combination. The Figure 9 shows how an example sensor configuration could look like. The configurator also offers to edit and delete existing activities.

Sensor A + Sensor B + Sensor C = cooking lunch

Figure 9: The activity “Cooking lunch” is defined by three sensors

Once the user confirms an action in the configurator, the data is sent via Ethernet to the “Mauser Runtime”, where the rule, depending on the action, is saved, edited or deleted.

3.11 System wide shut down

There is a physical push button in the environment, which is connected via “enOcean” with the “Mauser Runtime”. It is used for turning off the whole system in the environment. Upon pressing the button, an event is sent to the Runtime where the state of the system is checked. If the system’s state is off, it will be activated. Vice versa for the system deactivated state.

If the system is turned off, the controller of the “Mauser Runtime” ignores all incoming sensor signals. They are immediately discarded. Furthermore the Mauser Runtime informs

the smart TV application via Ethernet that in addition all incoming messages from the social community platform have to be dismissed. Apart from this message the “Mauser Runtime” also stops the communication with the smart TV application. The system stays off until the user reactivates the system by pressing the switch button. The “Mauser Runtime” receives the event from the “enOcean” automation bus, which is the only event that is not ignored, and from now on receives all incoming sensor signals.

4. References

[1] Image source: <http://www.enocean.com/en/radio-technology/>

[2] Image source: http://en.wikipedia.org/wiki/Radio-frequency_identification

[3] <http://www.samsungdforum.com/Guide/art00028/index.html>