Title:		Document Version:
Deliverable D4.10 **PLC-based Home Automation System Completed**		1.4

Project Number:	Project Acronym:	Project Title:
IST-2001-37613	6POWER	IPv6, QoS & Power Line Integration

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type* - Security**:
31/01/2004	04/02/2004	P – PU

Responsible:	Organization:	Contributing WP:
Antonio F. Gómez Skarmeta	UMU	WP4

Authors (organizations):

Jordi Palet (Consulintel), Laura Baño (UMU), Francisco José Herández (UMU), Inmaculada Marín (UMU), Daniel Martínez Manzano (UMU), Juan José Pujante Moreno (UMU).

Abstract:

This document describes the final home automation, featuring the integration of PLC technology with IPv6 devices.

Keywords:

Home automation, IPv6, OSGi, PLC, Services Gateway, UPnP.

Revision History

The following table describes the main changes done in the document since its creation.

Revision	Date	Description	Author (Organization)
v1.0	30/01/2004	Document creation	Daniel Martínez (UMU) Juan José Pujante (UMU)
v1.1	30/01/2004	Added appendix	Laura Baño (UMU) Inmaculada Marín (UMU)
v1.2	30/01/2004	Added hardware description of the X10 transmitter	Fco. José Hernández (UMU)
v1.3	30/01/2004	Final corrections	Daniel Martínez (UMU) Juan José Pujante (UMU)
v1.4	04/02/2004	Final review	Jordi Palet (Consulintel)

Executive Summary

This document describes the final home automation system developed by UMU in the context of the 6POWER project. This includes the integration of PLC technology with an IPv6 device for home automation and the deployment of automation services for remote monitoring and surveillance.

This automation system has been deployed in Zaragoza, as a part of Endesa's PLC network in that city.

Table of Contents

Table of Figures

1. INTRODUCTION

Deliverables D4.8 and D4.9 presented the design of a PLC based home automation system featuring remote control of both IPv6 enabled devices and simpler appliances, as well as the integration of PLC technology into home devices.

Work has been done to obtain a prototype of the final PLC based home automation system, which is described in this document.

This final home automation system comprises the UMU automation device introduced in D4.8 and D4.9, finally integrated with PLC, and services gateway based solution for the management of non IPv6 enabled devices, actually a lamp and a thermal sensor.

This automation system has been deployed in Zaragoza, as a part of Endesa's PLC network in that city.

A commercial service exploiting the results of this activity has been started by one of the project partners. A preliminary version was displayed in Brussels, during the Global IPv6 Service Launch Event, 15-16th January.

2. HOME AUTOMATION SYSTEM

This section shows the final developments related to the Home Automation System and its components.

We can identify hardware and software elements and new developments related to both kinds of elements.

The development described in this document has been done by University of Murcia. Other partners participated also in this activity, and indeed, Consulintel started a commercial service for automation, remote control and surveillance, in general, targeted to building constructors and alarm detection centers. A reduced version, with a real home with devices connected using several technologies (ADSL, PLC, WLAN, Ethernet, …) all with IPv6, was demonstrated remotely at the Global IPv6 Service Launch Event in Brussels, and described as impressive by Mr. Liikanen.

2.1 PLC-UMU Home Automation Device

The integration of the PLC PCI card in the UMU home automation device has been completed, achieving the goal established in the document [D4.9].

The UMU home automation device has now a main board with a PCI slot that allows fitting a PLC PCI card in the device. Thanks to this new feature the device no longer requires a network connector like an Ethernet port and one external PLC-CPE.

Due to the integration of the PLC PCI card and the home automation device, it will be possible to deploy these devices in places where no cabled networks are available.

Next, we can see some pictures of the final UMU home automation device prototype.
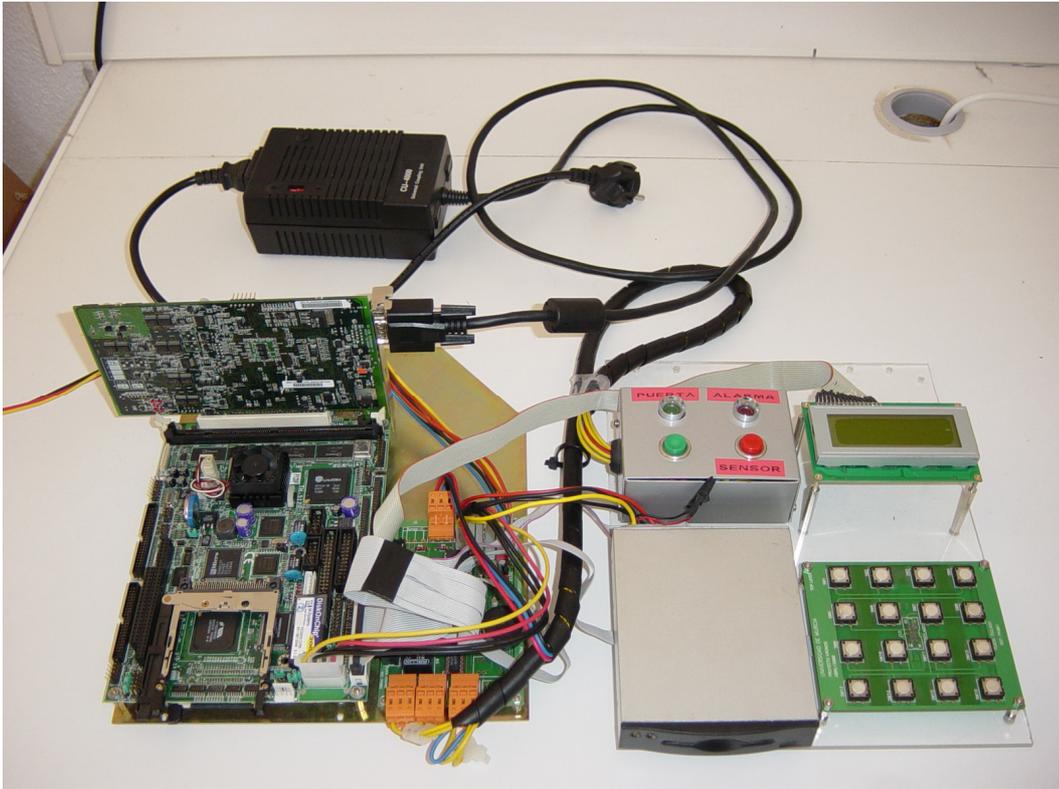
Figure 2-1: **PLC-UMU Home Automation Device**

In Figure 2-1 we can see the device and its PCI PLC card. Beside the main board are the I/O devices of the home automation device. These devices are an LCD display to show messages to the user, a keyboard matrix, the smart card reader and a box with lights to show when the door is opened and/or closed by means of the UMU device.

Figure 2-2 and Figure 2-3 show a more detailed view of the PCI PLC card and the home automation device's main board.

Figure 2-2: **PCI PLC Card and Device's Main Board (I)**

Figure 2-3: **PCI PLC Card and Device's Main Board (II)**

2.2 Technology Integration for Home Automation

It is a fact that technologies change and evolve throughout time. This way, new technologies arise, which eventually replace those becoming obsolete.

Such is the case of the X10 protocol, obsolete and slow today when compared to other newer home automation technologies like UPnP (Universal Plug and Play).

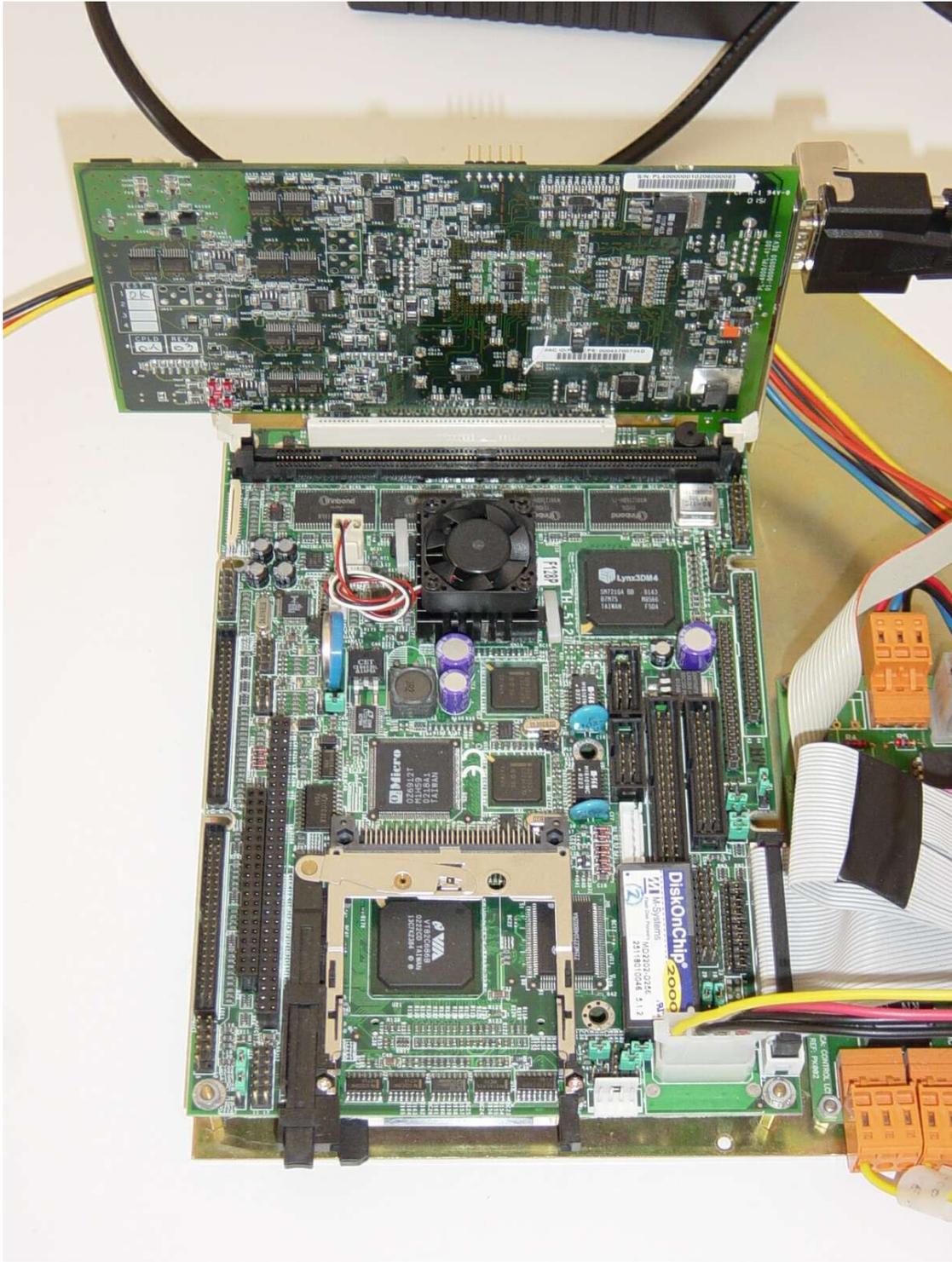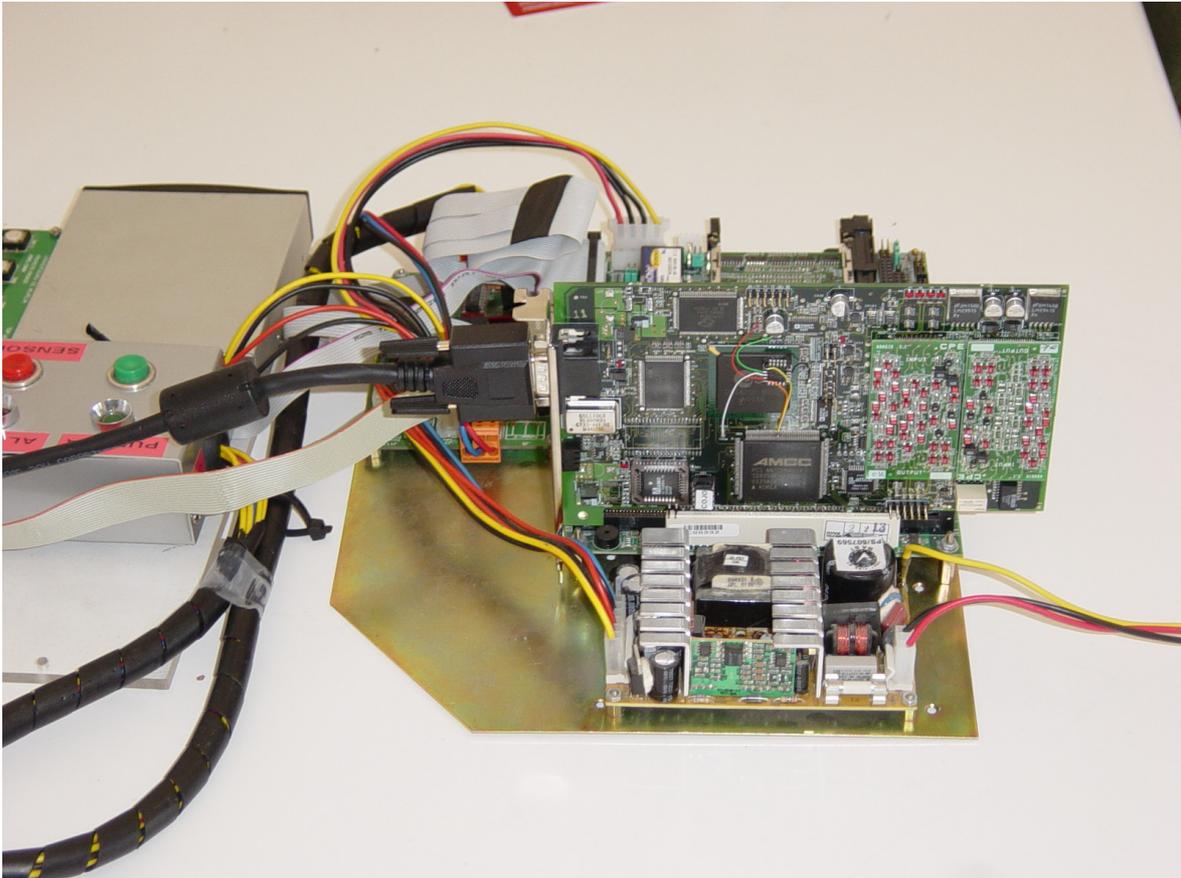For this reason, our proposal is aimed to the usage of a services gateway like OSGi, which makes transparent to the user the actual underlying technology which provides the service, even if it should change at any time.

As it was introduced in the [D4.8], the ideal proposed solution for a Home Automation System was the one where every device that might require a broadband connection can have a PLC interface to achieve this. This goal is achieved within the PLC integration in the UMU Home Automation device [D4.9].

Other desirable goal was to be able to manage non-IPv6 devices by means of a protocol/technology aggregator. OSGi was the presented as a solution for this too.

The OSGi functionality can be observed in the service developed by UMU that allows a user to manage X10 bulbs as IPv6 UPnP devices. The X10 technology is limited and not as powerful as it would be desirable. By means of the OSGi developed service, a user will be able to manage his X10 bulbs by means of any UPnP IPv6 Control Point.

In this case, the underlying technology is X10, but this is just an example of a non-IPv6 technology that is now accessible thanks to the OSGi service.

2.3   Deployment of the Home Automation Prototype in Zaragoza

As a result of the 6POWER Project, a home automation prototype has been deployed in Zaragoza, being a part of Endesa's PLC network in that city. It features remote monitoring and surveillance of a transformation center (TC). For this purpose, an IP camera has been installed inside the TC, as well as a OSGi-equipped computer for controlling a bulb. This way, it is possible to turn on and off the bulb inside the TC, as well as being able to watch for any anomalies inside the room. Moreover, a thermal sensor is included so that room temperature can be also managed via an OSGi service.

Following images show the automation prototype as actually deployed.

Figure 2-4:   Prototype Installed in the Transformation Center

Figure 2-5: **OSGi Server in the Transformation Center's Rack**

The installed light is using an X10 lamp adapter, whose control is managed via the aforementioned OSGi service. The control operations are executed by an X10 transmitter made by UMU. The board of this X10 transmitter is showed below.
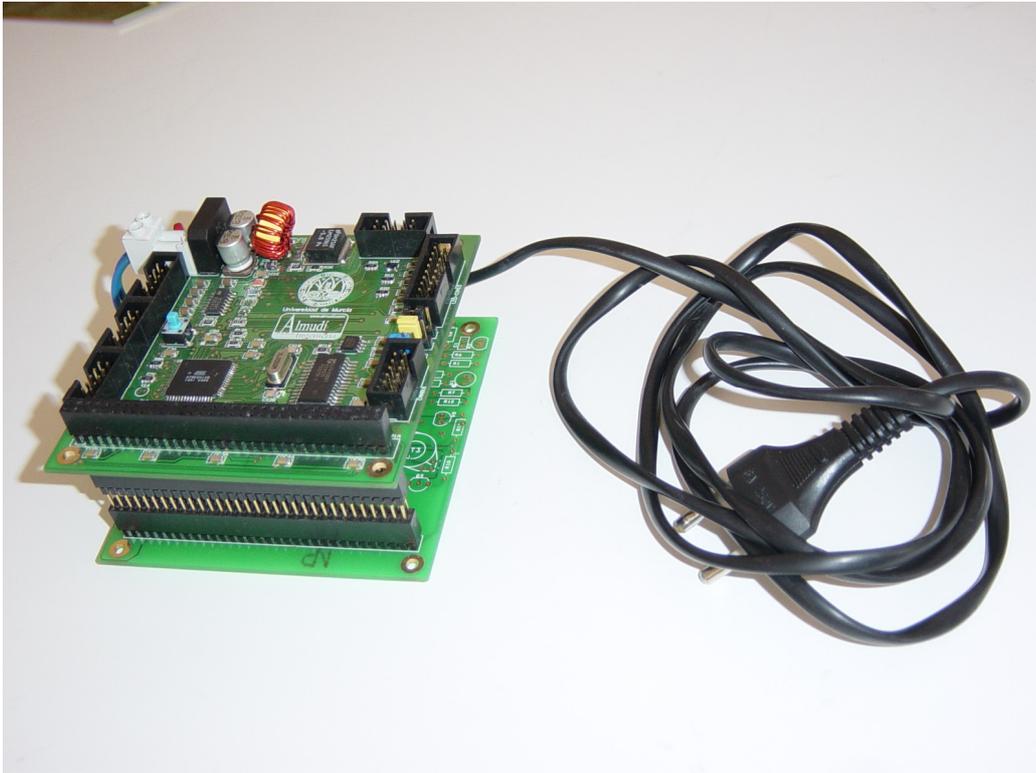
Figure 2-6:   UMU X10 Transmitter and Thermal Sensor

This device not only sends the X10 commands through the power line, it also features a thermal sensor for measuring indoor temperature. The temperature information is handled by another OSGi service, allowing obtaining further information about the monitored room. The interface between this device and the computer acting as OSGi server is a standard serial port.

This device is placed inside a small box located on the top of the rack inside the TC, as seen in Figure 2-4.

3. APPENDIX

3.1 UPnP-X10 Bridge Bundle

3.1.1 Introduction

Its objective is to integrate both X10 and UPnP technologies, taking advantage of the OSGi's aggregation/integration capability. With this, and having an X10 transmitter connected to a computer at our disposal, we will manage to create instances of X10 light devices that turn up as they were UPnP ones, interacting in an UPnP environment. Thus, any control point will be able to know about them, get their XML description documents, get access to their presentation pages in order to control them (invoking their actions and checking the value of any state variable which it is interested in) and subscribe to all the events generated by the services of these devices, due to changes in any of its state variables.

When the control point asks the UPnP device for changing its state, the UPnP-X10 Bridge will send the appropriate commands to the X10 light, via the transmitter. Then, the bulb will be physically switched on or off, according to the wishes of the control point.

3.1.2 Implementation

As said above, a simple light has been chosen as an X10 device. All it can be done with this simple device is switching it on or off. Then, we have an X10 transmitter connected to a computer and controlled by the serial port of that computer. This transmitter is the one developed by the University of Murcia, and it will send all the necessary commands to the bulb. Taking this into account, the UPnP-X10 Bridge Bundle consists of the following services:

- The first service is in charge of sending commands to the X10 lights through the transmitter, in order to switch them on or off, using their home code (from A to P) and device code (from 1 to 16). It also disconnects the transmission when the light devices are not needed any more.

- The second service implements a binary light device, which represents only one X10 light. This UPnP device is a binary light type, according to the standardized template approved by the UPnP Forum (urn:schemas-upnp-org:device:binarylight:1). This template defines a "simple representation of a bulb or any other kind of light emitting device that can be switched completely on or off" [UPNPF]; therefore, it perfectly adapts to our X10 light. It comprises of a Switch Power service, which is required, aimed at switching the device on or off. Nevertheless, due to the fact that possibly some X10 lights allow increase or decrease its light intensity, we have added a non-standardized service, called dimming, consisting of changing the intensity of the light source gradually. Although this functionality is not supported by our bulb, we have included it as a value-added service, in case we can make use of X10 devices supporting it.

- In addition, this UPnP device offers a presentation page, so as to let any control point know about the current state of the device and change it.

- The third service keeps essential information related to an X10 light (home code and device code) and its respective instance of an UPnP binary light device. When the status of the UPnP device is modified, this service sends the appropriate commands to the X10 device, using the first one.

- ▪ The fourth service creates, manages and keeps record of all instances related to the previous service. Thus, both services are proper the UPnP-X10 bridge.
- ▪ The last service implements a servlet that allows the generation, view, control and elimination of our own X10 devices, using the previous service. This one makes possible the integration of the technologies within OSGi.
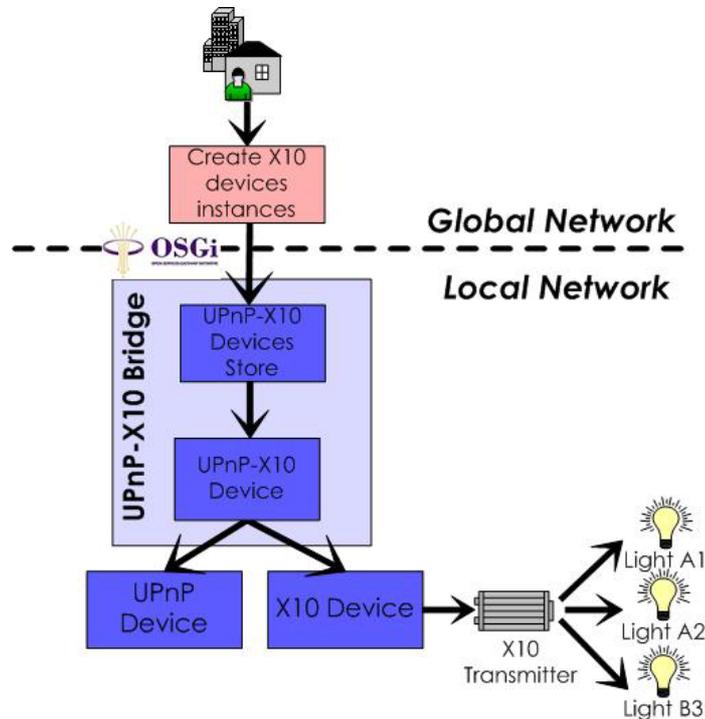

Figure 3-1: **Elements Participating in the UPnP-X10Bridge Bundle**

3.1.3 Operation

When using this bundle, we are shown a web page which let us create our local X10 devices by introducing some information, such as its unique identifier, friendly name, home code and device code. When creating an X10 light, an UPnP device is also generated. This device represents the new X10 light device we have just created. Once the UPnP device has been generated, it is announced within the local UPnP environment, so any control point can determine its presence. Then, it is possible for a control point to invoke the permitted actions over that device. As a result, the same actions will be invoked over the correspondingly X10 device via the transmitter, getting the physical bulb switched on or off.

By means of the previous web page, we can also control our X10 light device, changing its state and even stopping it and its associated UPnP device. This page also provides the chance of modifying the level of intensity of the UPnP device, without affecting the X10 one, provided that the latter does not support that functionality.

3.1.4 Requirements

In order to run this bundle, the CyberLink SDK together with Xerces is needed to make the UPnP device work properly. It is also necessary the X10 API for the X10 light device and the transmitter developed by the UMU. In addition to that, we obviously need the SMF framework to have a proper OSGI environment.

3.2 BinaryLights Bundle

3.2.1 Introduction

This bundle is aimed at making it possible to find all UPnP binary light devices existing in an UPnP environment. Then, a user will be able to choose one of them, observe its description and the description of the services it offers and gain access to its presentation page. Once there, that user will manage to know the state of the light device and change it in a remote way.

3.2.2 Implementation

The BinaryLights bundle makes use of the following three services:

- The first one consists of a service for a basic UPnP control point. It is in charge of searching for all UPnP root devices that are part of the local network.
- The second service uses the above one to obtain only devices of type binary light that are in the environment at present. That is, it gets the devices found by the previous service, retrieves those of binary light type and then keeps them in an internal vector.
- The third service is a servlet that receives requests from the web browsers. It will show the appropriate information, using the previous service.
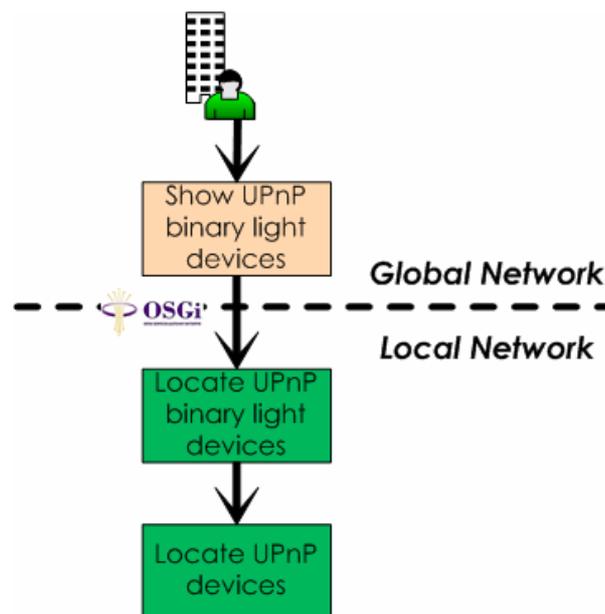
Figure 3-2: **Elements Participating in the BinaryLights Bundle**

3.2.3 Operation

When using this bundle, we can access to a web page that shows the existing UPnP binary lights devices in an UPnP environment, together with basic information, like their universal identifier and friendly name. By selecting one of these devices, we can have a look to detailed information about it, such as technical and descriptive one, the URL where its XML description document or presentation page can be found. It is also shown similar information about the services it offers. If we access to its presentation page, we will know if the binary light is switched on or off. Besides, we can change its state. What is more, if the chosen UPnP device represents an X10 light, those state changes will affect it too, as commented above.

4. SUMMARY AND CONCLUSIONS

The integration of PLC technology with IPv6 home automation devices allows for end-to-end remote monitoring, control and surveillance reaching places where no cabling exists other than the power wires. Besides, services gateway based solutions like the one presented in this document favor firstly the integration of different technologies under a common interface, and secondly the independence of that common interface from the variation and evolution of technologies.

The results of this activity have already been included in the exploitation of a new commercial service by one of the partners, and had a highly impacting visibility during the past Global IPv6 Service Launch Event.