Plug and play your robot into your smart home: Illustration of a new framework

Sao Mai Nguyen, Christophe Lohr, Philippe Tanguy, Yiqiao Chen

Plug and Play your Robot into your Smart Home: Illustration of a New Framework

A Communication Protocol to Plug and Play any Robot or Device of a Smart Home

Sao Mai Nguyen · Christophe Lohr · Philippe Tanguy · Yiqiao Chen

Abstract We present our team IHSEV, and our preliminary studies to tackle the question of interoperability of devices and robots for smart homes. We propose a framework enabling the seamless communication between smart home devices and robots. Our framework relies primarily on the xAAL protocol, which allows any device from any type to be plugged into a smart house network. We have recently extended xAAL to allow any ROS-compatible robot to be integrated into the smart house network. We illustrate the relevance of this framework in an implemented use case: assistance of an elderly person in the case of a fall (Fig. 1).

Keywords Home automation networks · Interoperability · Ambient assisted living · Service robots · Agent-based smart home · Robot-assisted future home environment

1 IHSEV team

1.1 Goal and Purpose of the IHSEV Team

A significant increase in life expectancy over many decades has given rise to concerns about the quality of life for the elderly. Statistics in different countries report that seniors just want to stay the long-term in their homes. The European Union is inciting innovation and research projects enabling independent living for elderly people. The IHSEV (French initials for *Human-Systems Interactions and Virtual Environnements*) team claims that ICT may be able to alleviate this situation. We are evaluating the advantages and drawbacks of using humanoid robots as *helpers* for people with loss of autonomy. We explore the technical solutions for a seamless monitoring and assistance of the users through a network of sensors and actuators, combined with a natural and social interface and mobile systems. Thus we propose the ensemble of a robot interacting with a smart home, for a more comprehensive and interactive assistance to users in their homes.
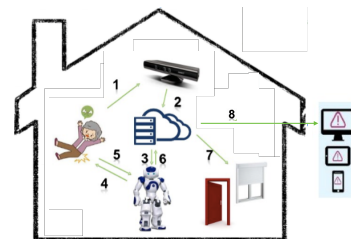

Fig. 1: Assistance in the case of fall: relevance of the interoperability of home automation devices and robots

1.2 Smart Environment Research Installations

In order to study the impact of new technologies introduced on users' everyday lives, we have chosen to carry out our user studies in home environments. The Living Lab infrastructure "Experiment'HAAL" (Fig. 2) has been set up to facilitate the co-design of technical and real experimentation in order to analyse their interactions and their uses. It consists of three rooms 70 m^2 in total, including an apartment, a desk for professional users and an observation room. The Experiment'HAAL lab is used for test campaigns, in the context of ambient assistive living, telemedicine and tele-care, together with sociologists and ergonomists.

Sao Mai Nguyen
Institut Mines Télécom Atlantique and Lab-STICC
E-mail: nguyensmai@gmail.com

Christophe Lohr
Institut Mines Télécom Atlantique and Lab-STICC
E-mail: christophe.lohr@imt-atlantique.fr

Yiqiao Chen, Philippe Tanguy
Institut Mines-Télécom Atlantique

Fig. 2: Experiment'HAAL living lab : map of the apartment and pictures

Available devices and sensors include: access control entrance, door sensor entrance, fridge, drawer cabinet, cupboard, smoke detector, presence detector, pressure sensor armchair, convertible chair bed, water flow sensor washbasin, sink, water leak detector sink, wireless remote controlled outlet, convectors with wire pilot management, power consumption sensor and a sense floor covering the entire apartment.

1.3 Integrating IoT and Robotics

Nowadays, smart homes are no longer a far-off dream. They are becoming affordable to ordinary people. Its goal is to increase the comfort and security of our homes. Such developments would allow the elderly to stay in their cosy homes, and avoid their reluctant moving to care homes away from their families and social environment. They could be part of a solution to the societal challenge of the ageing population.

However, current smart home solutions present limitations since the sensors and actuators are most of the time only capable of carrying out binary actions and of undertaking simplistic tasks. Second, this network of sensors and actuators are set to fixed locations covering a predefined area, which means limited sensing ranges. A mobile robot allows access to uncovered areas in a habitat. It allows the dynamical addition of other points of view (more information) to provide to algorithms trying to make a decision in order to to assist a user in his/her daily life. In other words, the combination of data from a static sensors/actuators nodes (classic home automation devices) and a mobile node (the robot) offers new capabilities to improve ambient assisting living services.

At the same time, service robotics also promise a better comfort for the disabled and the elderly, assisting them in their everyday tasks and improving their social interactions. Compared to the current smartphones, connected televisions or tablets, humanoid robots will allow a more natural and intuitive interaction for the users, but also a fuller interaction owing to the multi-modal interaction a robot can engage in: vocal, tactile or visual interactions.

Albeit mobile, as the sensors and actuators of robots are located at the same spot, their range of action and sensing is limited to their physical location. They could not carry out actions in other rooms. For instance, if the robot were cooking in the kitchen, it would not be able to detect falls of a person in a living room. Connecting the robot to a network on sensors and actuators could thus address the shortcomings of each system into a complete solution capable of both gathering information seamlessly, carrying out actions at any locations, undertaking complex task and interacting in a multi-modal and social manner. We argue that a robot connected to a smart house could bring about relevant solutions to improve comfort and security for the elderly.

2 State of the Art: Interoperability of Smart Devices

The emergence of Internet of Things (IoT) and smart-home systems allows us to combine devices from different domains and to explore new uses and services. Unfortunately, interoperability between devices from different technologies is a major issue to overcome before being able to offer smart services.

2.1 Home Automation Interoperability

One of the main issues in home automation systems is the choice of a communication protocol. Among classical protocols (Z-Wave, KNX, EnOcean, etc.), some of them are widely used; however, their interoperability is not well considered. For instance, if someone wants to use a sensor using Z-Wave with an actuator using KNX, it is not feasible without a special gateway in charge of the interoperability between both protocols.

Interoperability solutions face several challenges. First of all, sensors and actuators do not have the same characteristics of the physical layer and of the transmission channel in use (frequency, throughput,

bandwidth, data coding, etc.) On the other hand, sensors and actuators are dependent on application domains. Finally, the business model has a deep impact on the system architecture currently deployed in a house. The consumer is facing a captive market: few providers with solutions that cover every needs, with few chances to add equipment from others. That is why the common architecture is composed by a network of sensors/actuators speaking the same proprietary protocol and a gateway gathering all the messages. The gateway also plays the role of manager with an administration interface and offers a connection with Internet to develop access via a mobile application. The main challenge is the interoperability between devices. Actually, there is a lack of standardization in the home automation systems. In the past 10 years, a lot of initiatives, alliances, or standards appeared, like ZigBee, EnOcean, or Z-Wave; however, they still do not promote interoperability with other alliances or standard. Existing solutions can be classified in three categories: protocol of communication, middleware, and architecture/framework. For those three categories, common mechanisms (or functionalities) are used to achieve interoperability. Indeed, most of the solutions have similar components or functionalities that are essential for a system including heterogeneous devices: addressing schema for nodes, description or semantic of participating nodes, discovery mechanism, communication protocol, security and privacy, etc. To sum up, almost all solutions play around the idea of a so called *gateway*: a box (a small computer) is equipped with two or more modules, each module talking in a given home-automation protocol; then, a piece of software over those modules make it possible to forward messages between protocol A and protocol B. Moreover, those boxes embed additional functions to make all of these usable: HMI (via Web and/or smartphones), configuration facilities, connectors to extra services in the cloud, etc. [7] presents a review of existing solutions. One of the most advanced solution is probably openHAB [14]: a modular expandable open-source gateway implemented in Java. However, this is also a centralized solution. Such solutions are partial from the interoperability point of view, as each box manages a limited number of low-level protocols, and boxes are hardly able to talk to each other. To overcome those limitations, the xAAL system have been proposed [7]. Basically, the idea is to reorganize and formalize the architectures of those boxes, which are in fact all composed of the same functionalities. The xAAL system relies on a distributed architecture of functional entities, communicating to each other via a message over IP.

2.2 Robots Interoperability

The past 20 years have witnessed the development of numerous robotic platforms geared towards service robotics. Nevertheless, it soon appeared that problems that could seem trivial to humans often are very hard to solve, because of the variability of the conditions, environment or instances of tasks. Dealing with these variations is so hard that no single laboratory or institution can hope to do so by themselves. Thus, a need to be able to plug different functionalities/abilities to enhance the potential of a robotic platform arises. To enable this, the RobotCub Consortium, a consortium of several European universities, decided to share the same humanoid robotic platform [10]: the iCub [12]. Nevertheless, most laboratories and universities researching in the field of robotics are still developing on very different platforms. The interoperability of the various functionalities developed by each laboratory or institution on their own robotic platform is then crucial.

Besides, a collection of software frameworks for robot software development, was built to encourage collaborative robotics software development. The Robot Operating System (ROS) [16] is a framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating a complex and robust robot behaviour across a wide variety of robotic platforms. This middleware provides standard operating system services such as hardware abstraction, low-level device control, implementation of commonly used functionality, message passing between processes, and package management. Currently, ROS has been made compatible with dozens of robots and ports, enabling integration of different functionalities into a single robot, and enabling communication between different robots [13].

While efforts have been undertaken to enable interoperability in the domains of robotics and home automation, few open solutions have enabled the interoperability of both. The *iot_bridge* package [15] provides a bridge between ROS and openHAB, allowing a ROS application to interact with a given open openHAB box. Actions are limited to home-automation devices managed by the given box. We can also cite the SmartSEAL project which proposes a home-automation system by reusing the ROS infrastructure as technical basement [3]. Although the authors do not use robots in their system, SmartSEAL is probably compatible with robots developped with ROS. Our solution is slightly different since this allows a robot to be seen as a home-automation device. We will illustrate our solution for home automation-robot interoperability, in a use-case
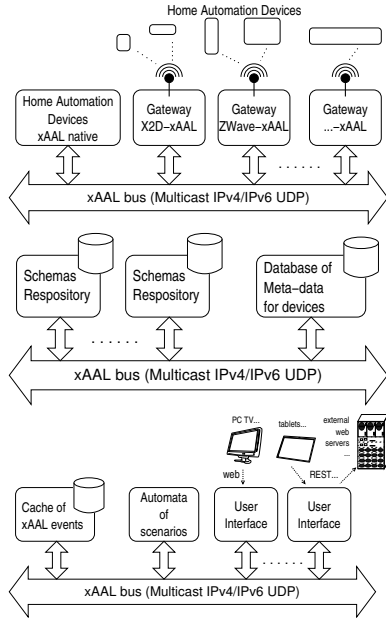
Fig. 3: Functional Architecture of xAAL.

after a description of the architecture of the xAAL-ROS system.

3 Robots - Smart Home Interoperability Based on xAAL

3.1 xAAL: A Distributed Infrastructure for Heterogeneous Ambient Devices

To overcome interoperability issues in the home automation domain, a sensors-actuators layer named xAAL has been proposed [7,8].

xAAL is a functional distributed architecture; all components talk to others via an IP (multicast) bus. The communications are in the form *many-to-many*. Messages are in JSON format ciphered with Poly1305 Chacha20. The set functions traditionally implemented by home-automation boxes are arranged into well-defined separated functional components: elementary gateways for technology-specific devices, native-xAAL devices, caches, configuration database, user interfaces, scenarios automata, etc. Each component may have multiple instances, may be shared by several boxes, and may be physically located on any box. Figure 3 depicts the general architecture of xAAL.

A device has a type, described by a schema. The schema provides semantics to a device and describes its capabilities: a list of possible methods on this device, a list of notifications that it sends spontaneously, and a list of attributes Those schemas are inspired by the UPnP approach [6].

Messages are transported by an IP multicast bus (IPv4 or IPv6). As proposed by UPnP, a bus allows the discovery: when a new component appears in the installation, it announces itself. All other entities can then take it into account. Similarly, when a new component enters, it can query the bus to discover the other components already present. This greatly facilitates the configuration and allows dynamicity.

3.2 Architecture of the system based on xAAL and ROS

To enable interoperability with most robots, we designed an architecture based on xAAL and ROS. The components of this illustration are shown in Fig. 4a. They include devices using xAAL such as automated entrance doors, electric plugs and the telephone. They also include a ROS-compatible service robot, which can have voice recognition and speech functionalities, a camera, etc. We can also link our system to other programming interfaces for communication with the systems or people outside the smart home. In our implementation, we used the ROS package of WebRTC, a JavaScript programming interface used by various web browsers for real-time communication. The goal of WebRTC is to link applications such as video streaming, voice over IP, peer-to-peer file sharing.

As illustrated by Fig. 4b, the computational architecture of this system is based on a central controller connected to ROS nodes for the robot or other components such as the Kinect camera, and the xAAL proxy that manages communication with home automation devices. The central controller is also connected to other classes, in order to communicate by email for instance. The central controller is thus both a ROS node and a xAAL component. It analyses the collected data and insures the high-level functions of the system.

3.3 Requirements and Properties

As our system relies on xAAL and ROS, it must comply with requirements of both. xAAL bears implementations in C, Python and Java and is operational under Unix-like system as well as Windows, whereas ROS is geared toward Unix-like systems and are mainly proposed in C++, Python and Lisp. Therefore we have chosen an implementation of our ROS node for linking xAAL and ROS in Python, using a Unix system.

Contrarily to former solutions that rely on a centralized system, owing to the design of xAAL, the proposed solution allows to jump in the world of distributed systems and takes benefits of it.

(a) Representation of the different components used in out illustration use case.
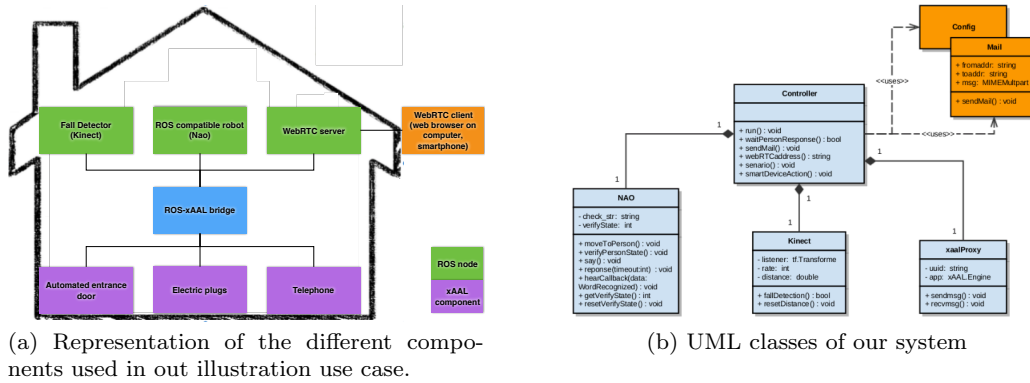
(b) UML classes of our system

Fig. 4: Architecture of the system based on xAAL and ROS

Cybersecurity-wise, our implemented system has not tackled this issue. Although ROS has no authentication system or security layer, some recent extensions bring security for network communication between robots components. For instance, the experimental SROS [17] adds a set of security enhancements, such as native TLS support for all socket transport. Likewise, xAAL has added a security layer since v0.5. Tackling security issues could be future work.

Both xAAL (`http://recherche.telecom-bretagne.eu/xaal/`) and ROS are open-source, thus we decided to share our implementation on `https://github.com/yiqiaoc/ROS_Package`. We implemented a use-case for assistive technologies that is described in the following section.

4 Use Case: Assistance after a Fall at Home

We wished to illustrate the potential of a comprehensive system based on both a smart home and a mobile robot in a home assistance scenario. For this illustration, we tried to tackle one of the leading cause of injury deaths [5] and of injury-related hospitalisation [2] among the elderly population: falls affect over one in every three elderly people [9]. Assistance of the elderly in the case of falls are thus crucial for their everyday safety.

4.1 Challenges for Assistive Technologies in the Case of a Aall

However, a lot of challenges remain unsolved on this topic. First of all, systems for fall detection are still unreliable, especially for detecting falls of the elderly at home, despite many efforts [1,4,11]. Second, as the person who fell down can no longer move or call for help or with their family, they are often in a state of distress. They could also face dangers as they are no longer able to control ongoing processes in the house: turn off water

taps, the cooker or the oven. Even in the case their fall is detected and rescue services could arrive, they often have to break open the entrance door or a window to access to the victim.

We wished to illustrate how a system integrating a network of smart devices and robots could help tackle current issues for assistance in the case of falls. We propose in the next section an implementation of a an assistance scenario based on the interoperationality of smart home devices and robots, highlighting the complementarity of the home automation and robots.

4.2 Illustration Scenario

To tackle the stated issues, we propose to illustrate a system in our living lab capable of:

1. Detecting a fall. In this demonstration, we used a kinect rgb-d camera, and compared the heights of the hip and head to simplistically detect a fall.
2. Performing a confirmation of the fall, to alleviate the reliability problem of the fall detection system. This confirmation relies on speech interaction with a social robot.
3. Securing the house owing to actuators in the house. We used the humanoid robot Nao (by SoftBank Robotics).
4. Facilitating communication with the family or rescue services, through both telephones or robot-embedded cameras.
5. Allowing access to the house for rescue, owing to automated door locking.

We built our system along the scenario proposed in Fig. 5. As soon as a probable fall has been detected by the fixed sensors, we proceed to the confirmation for reliable information. A social robot interacts with the user orally to confirm the fall. If the fall detection was erroneous, it would return to its normal state. If the fall indeed occurred, it could provide help to the
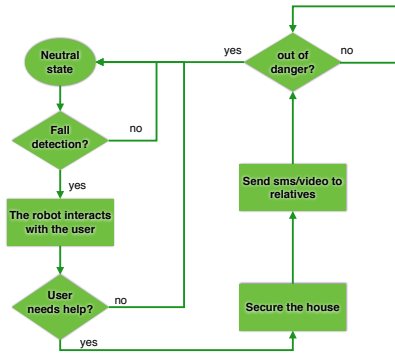
Fig. 5: State graph for the proposed scenario of assistance in the case of fall.

victim. We can imagine that it could bring in objects, medicine, etc. In our scenario, it first secures the house by turning off all ovens, cookers, running water taps. It can dial automatically a number to allow the victim to speak to a relative. It also sends sms and emails to the relatives to enable them to confirm and reassure the victim. In our scenario, the email includes an ip address that would enable them to see the video flow from camera of the robot.

5 Conclusion

The described system calls for improvement for each of the functionalities used. To cite a few limitations, the fall detection system presented in this paper is very simplistic and requires the user to be seen fully by the Kinect camera. Besides, the fall confirmation by the relatives we propose consists only in the visualisation of a video stream. We could improve this passive solution by giving the relatives the possibility to control the robot so as to move the robot to relevant locations for a clearer point of view. Nevertheless, the described use-case showcases a seamless integration of IoT and robots for smart home automation interoperability. We argue that facing the numerous home automation devices and the numerous robots developed, but also due to the variability of each home and the variability of the users' use, we need a plug-and-play system that can integrate any robot and any smart home device. Our preliminary open-source work should reduce time/effort required to setup a variety of components for experimentation. We advocate for a comprehensive solution of assistive technologies including robots and smart homes in order to bring out the advantages of both. In future works, we wish to improve our sytem by testing with a more comprehensive home automation system in terms of reliability. We will also analyse the specifics of the roles of robots and home automation systems in terms of sensors, actuators but also interaction.

References

1. Human fall detection on embedded platform using depth maps and wireless accelerometer. Computer Methods and Programs in Biomedicine **117**(3), 489 – 501 (2014)
2. Baker, S.P., Harvey, A.H.: Fall injuries in the elderly. Clin Geriatr Med **1**(3), 501–12 (1985)
3. Bellocchio, E., Costante, G., Cascianelli, S., Valigi, P., Ciarfuglia, T.A.: Smartseal: A ROS based home automation framework for heterogeneous devices interconnection in smart buildings. In: IEEE International Smart Cities Conference, ISC2 2016, Trento, Italy, September 12-15, 2016, pp. 1–6 (2016)
4. Chan, A.M., Selvaraj, N., Ferdosi, N., Narasimhan, R.: Wireless patch sensor for remote monitoring of heart rate, respiration, activity, and falls. In: 2013 35th Annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp. 6115–6118. IEEE (2013)
5. Hoyert, D.L., Kochanek, K.D., Murphy, S.L.: Deaths: final data for 1997. Natl Vital Stat Rep **47**(19), 1–104 (1999)
6. UPnP Device Architecture—Part 1–1: UPnP Device Architecture Version 1.1. Standard, International Organization for Standardization, Geneva, CH (2011). `https://www.iso.org/standard/57494.html`
7. Lohr, C., Tanguy, P., Kerdreux, J.: xaal: A distributed infrastructure for heterogeneous ambient devices. JISYS **24**(3), 321–331 (2015)
8. Lohr, C., Tanguy, P., Kerdreux, J.: Choosing security elements for the xaal home automation system. IEEE Proceedings of ATC 2016 pp. 534–541 (2016)
9. Lord, S.R., Ward, J.A., Williams, P., Anstey, K.J.: An epidemiological study of falls in older community-dwelling women: the randwick falls and fractures study. Aust J Public Health **17**(3), 240–5 (1993)
10. Metta, G., Sandini, G., Vernon, D., Natale, L., Nori, F.: The icub humanoid robot: An open platform for research in embodied cognition. In: Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems, PerMIS '08, pp. 50–56. ACM, New York, NY, USA (2008)
11. Narasimhan, R.: Skin-contact sensor for automatic fall detection. In: 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 4038–4041. IEEE (2012)
12. Official iCub website: An open source cognitive humanoid robotic platform. `http://www.icub.org/`, Accessed 7 April, 2017
13. Official ROS website: Robot operating system. `http://www.ros.org`, Accessed 7 April, 2017
14. openHAB: openhab project. `http://www.openhab.org/`, Accessed 7 April, 2017
15. openHAB add-on: iot_bridge. `http://wiki.ros.org/iot_bridge`, Accessed 7 April, 2017
16. Quigley, M., Conley, K., Gerkey, B.P., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y.: Ros: an open-source robot operating system. In: ICRA Workshop on Open Source Software (2009)
17. SROS website: Sros - ros wiki. `http://wiki.ros.org/SROS`, Accessed 27 April, 2017