

# Enabling new e-health business models by converging IP-based and non-IP-based home networks

Frank den Hartog, Martin Tijmes, Eelco Cramer  
TNO  
Delft, The Netherlands  
[frank.denhartog, martin.tijmes, eelco.cramer]@tno.nl

Arjen Noorbergen  
Quby  
Amsterdam, The Netherlands  
arjen.noorbergen@quby.nl

**Abstract**— Current efforts in extending the Internet are focused on achieving full support of services and user-centric content from sectors such as energy management, education, and e-health. Until recently, these sectors typically developed their own communication infrastructure and systems, leading to point solutions, often built on non-IP technologies. We describe a demonstrator that shows how a non-IP supporting Continua-certified health device, such as a weighing scale, can be unlocked for use by other devices and services in the home by applying Universal Plug and Play (UPnP), and how this enables new use cases and business models.

## I. INTRODUCTION

The future Internet will not be restricted to the IT, telecom and media sectors. Current efforts in extending the Internet, such as undertaken by the European Future Internet Alliance, are heavily focused on achieving full support of services and user-centric content from other sectors, such as energy management, education, and e-health. Moreover, users will expect to access these services and their content in the same manner as any other Internet-based service. In the context of the home, this implies that content and functionality can be freely shared between devices, irrespective of the service for which the device is originally designed for. For instance, a single tablet-like display can then be used for watching managed IPTV services, but also for having a video communication with the district nurse, and controlling the temperature in the room.

In the past, these sectors typically developed their own communication infrastructure and systems, leading to point solutions, often built on non-IP technologies. Hence, there is a need for coordination across different sectors in order to interconnect system and have them collaborating. From a technical perspective, the home gateway is in a good position to take such a coordinating role among the various devices in the home and their Internet connectivity.

The European FP7 FIGARO project ([www.ict-figaro.eu](http://www.ict-figaro.eu)) researches various alternative technical solutions for achieving such convergence. In this paper we describe a demonstrator that shows how non-IP supporting Continua-certified health devices can be unlocked for use by other devices and services in the home by applying Universal Plug and Play (UPnP), and how this enables new use cases and business models. In the following section we elaborate on these use cases and business models, after briefly introducing

Continua and UPnP. In section III we then describe the architecture and operation of our demonstrator.

## II. CONVERGING CONTINUA WITH UPnP

The Continua Health Alliance ([www.continuaalliance.org](http://www.continuaalliance.org)) aims to establish an interoperable ecosystem for personal health systems by selecting standards and specifications, and define these in guidelines. It considers the complete e-health end-to-end chain from measurement devices in the home (such as blood pressure meters, glucose meters, and scales) to the clinical information systems at the health care provider's premises. The Continua Design Guidelines v2010 require health devices to communicate with a so-called Application Hosting Device (AHD) by means of Bluetooth (BT), Zigbee, or USB. The AHD is a data concentrator in the home that collects the measurement data from the Continua-certified health devices, converts this data from the required ISO/IEEE11073 format to HL7 (an application layer protocol that is readily used by clinical information systems), and sets up a protected IP connection to the medical back-end of the user's health care provider. The AHD is not designed for sharing the measured data directly with the user. The user may or may not be able to access his data via a proprietary website provided by the health care provider.

UPnP ([www.upnp.org](http://www.upnp.org)) is a client/server-based interoperability framework for devices in, typically, a home network. It distinguishes UPnP Services, which represent the service functionality or capabilities of a device, UPnP Devices, which act as services servers, and UPnP Control Points (CPs), which act as clients for controlling the services. For clarity, UPnP Devices are not physical devices, but the logical containers of UPnP Services running on such devices. The UPnP Device Architecture defines a range of protocols for discovery, control, and eventing between devices. Device and Service descriptions are expressed and partially standardized in eXtended Markup Language (XML) templates called Device Control Protocols (DCPs). The UPnP forum is in the process of developing DCPs for, amongst others, energy management and e-health services.

We propose that Continua AHDs are extended with UPnP and an Application Programming Interface (API) between the UPnP stack and their native ISO/IEEE11073 stack (see Figure 1). New use cases and business models then become possible. For instance, third parties may provide

advanced visualization software for users to record and contemplate their health data at home. Or health insurance companies may provide advice services or reduction on the premium if customers can show their engagement in staying healthy, by sharing data they choose to provide. This business model, in a non-automated way, has already proven to be successful in the UK and South Africa [1]. The use case we are showing with our demonstrator is a variety on this. Further use cases become possible if the AHD also provides an UPnP API towards the HL7 interface. Health care providers can then access UPnP devices such as cameras and printers in the home to offer additional value to recording the data from Continua devices.

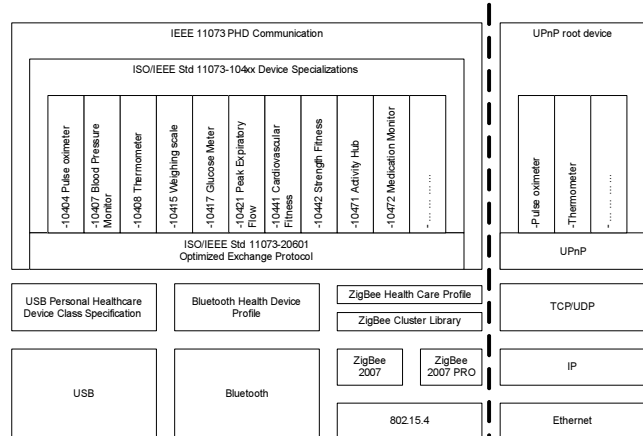


Figure 1. Proposed architecture of a Continua/UPnP proxy for AHDs

Our demonstrator (see Figure 2) is centered around a Continua-certified weighing scale. In the UPnP network there is a Continua/UPnP AHD proxy and a BT/UPnP proxy [2,3], both implemented on a Cotton Candy USB stick size PC. A Samsung Galaxy Tab 10.1 Android tablet PC and a laptop with an SMS gateway acting as a game server are UPnP Control Points. In the BT network there is a light and a watch. A known person stands on the scale. Using the UPnP backbone and the proxies, his weight is transferred to his BT watch. Bystanders or other conference attendees cannot see this. Instead, they can become game participants by registering themselves on the game server. When the person stands on the scale, the BT lamp lights up, and participants will receive a text message on their phone, inviting them to guess and text back the weight of the known person.

### III. DEMONSTRATOR ARCHITECTURE AND OPERATION

The weighing scale is the Continua-certified Omron HBF-206BT. After pressing a button, the scale sets up a Bluetooth connection with our AHD proxy using the BT Health Device profile. It then transfers the measured weight to the AHD using the ISO/IEEE11073 data format. The Continua/UPnP AHD is implemented using the Antidote IEEE11073:20601 open source health daemon from Signove [4]. The AHD itself is build on top of Node.js (http://nodejs.org). We have used, and contributed to, an open source UPnP device library for creating the UPnP device [5]. The AHD communicates with the Signove Continua health daemon using the DBUS message bus [6].

The Continua/UPnP proxy can be implemented in different ways, demanding different levels of intelligence from either the proxy or the CP. One option is to have the proxy discovered as a UPnP Continua Proxy Device. It exposes a proprietary user interface, which the CP then uses to access individual Continua-certified devices in the Continua network. This demands specific intelligence (apps) on the CP. We chose to have the AHD exposing every single Continua device in the Continua network as individual UPnP Devices, e.g. a to-be-standardized UPnP Weighing Scale. This demands less intelligence from the CP and guarantees a higher degree of interoperability in the home network. However, it also requires frequent updates of the AHD's firmware, whenever new UPnP Devices are standardized. For this, UPnP Device Management could possibly be used. This is for further research.

The BT/UPnP proxy and lamp are implemented as described in [3], as is the UPnP CP on the Android tablet. The BT watch is a Pebble E-paper Watch with customizable user interface. When the scale, light, and watch are switched on first, they register themselves as UPnP devices with the CP on the tablet. A known person writes his name in the game server, and stands on the scale. A GENA event containing the weight is then broadcasted in the UPnP network. The CP consecutively switches on the light, reads the data from the event message, and sends this data to the watch and the game server. The game server sends a text message to the participants and collects the replies.

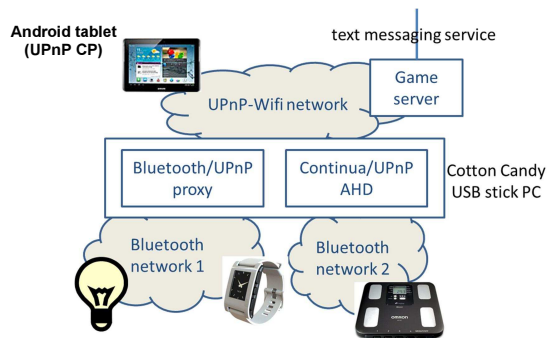


Figure 2. Demonstrator set-up

### REFERENCES

- [1] Deepak N. Patel *et al*, "The Association Between Medical Costs and Participation in the Vitality Health Promotion Program Among 948,974 Members of a South African Health Insurance Company", *American Journal of Health Promotion*, 24 (2010) 3, pp. 199-204.
- [2] A. Delphinanto *et al*, "Architecture of a bi-directional Bluetooth-UPnP proxy", in *Proc. of the 4th IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV (USA), 2007.
- [3] Gerard Mekenkamp *et al*, Deliverable 5.3: "Evaluation of converged gateway-centric, UPnP-based FI-architecture for energy management, e-Health, and social community services", [www.ict-figaro.eu](http://www.ict-figaro.eu) (2011).
- [4] [http://oss.signove.com/index.php/Antidote:\\_IEEE\\_11073-20601\\_stack](http://oss.signove.com/index.php/Antidote:_IEEE_11073-20601_stack)
- [5] <https://github.com/jacobrask/node-upnp-device#readme>
- [6] <http://www.freedesktop.org/wiki/Software/dbus>, <https://github.com/Shouqun/node-dbus#readme>