

UbiPAN: A Bluetooth Extended Personal Area Network

J r mie Albert, Tegawend  F. Bissyand , Y rom-David Bromberg, Serge Chaumette and Laurent R veill re

Department of Computer Science - University Bordeaux I
LaBRI, Computer Science Laboratory
351 cours de la Lib ration
F-33405 Talence Cedex, France
firstname.lastname@labri.fr

Abstract—Most mobile devices are now Bluetooth-enabled. This wireless technology makes it possible to transfer files or stream contents between pieces of equipment. The possibly many devices that are in reach of each other and thus connected constitute a Personal Area Network (PAN). The problem remains that because of radio range setting up such connections is only possible for devices in a limited area.

This paper presents the UbiPAN network infrastructure, the goal of which is to overcome this limitation. It relies on the combination of the IP network, SIP and Bluetooth. This original approach thus has been validated through a file exchange scenario between remote devices that have been seamlessly interconnected.

Keywords-Ubiquitous computing, Bluetooth, extended PAN, Gateway, SIP

I. INTRODUCTION

Context. Enabling ubiquitous computing requires to interconnect the many devices which populate our daily environment. This can be achieved today. The latest technological breakthroughs in wireless networks thus participate in fulfilling Weiser's vision of computing for the 21st century [1]. Devices are now dynamically connected to each other using wireless communication technologies such as Bluetooth [2], WiFi [3] or Infrared.

Unfortunately, these technologies present some limitations that hinder their adoption as reliable solutions for interconnecting devices in a large scale environment. For instance, Infrared is known to be inappropriate but for specific pieces of hardware, because a direct line of sight is required between the devices to be connected. WiFi may require heavy configurations and it is therefore more adapted for equipments in areas known in advance rather than for personal devices used in highly mobile ambient environments.

Bluetooth is increasingly adopted by electronic manufacturers as the primary inter-device communication technology. It is widespread because of four main reasons: (i) it is a standard (IEEE 802.15), (ii) the device is cheap, (iii) it is a low power consumption technology and (iv) its transceiver is physically small. It can thus be incorporated in any domestic or office electronic device. However, it shows severe limitations in terms of communication range, most Bluetooth-enabled devices having only a 10m communication range. The question is then how to extend seamlessly the range of Bluetooth?

Current solutions. Two major solutions have emerged so far to address this range issue: access points and repeaters.

Access points provided by manufacturers such as *Inventel* or *Belkin* relay data between wireless devices and the wired network. Repeaters amplify the radio signal and retransmit it. Both solutions provide wider coverage than basic Bluetooth. However, none of them allows a seamless interconnection of Bluetooth devices. Furthermore, to cover a large area, several access points or repeaters must be deployed and this approach thus does not scale because the number of nodes required is proportional to the distance between the devices to interconnect.

Our approach. This paper introduces a novel approach to seamlessly extend the communication range of Bluetooth-enabled devices. This approach does not require any dedicated firmware to be deployed on the involved equipments. To illustrate this, we consider the following scenario. Assume that you are attending a conference with colleagues attending a conference. You can take pictures and transfer these to their mobile phones using Bluetooth, provided they are in range of your own Bluetooth phone. Now assume your TV is either equipped natively or supplemented with a Bluetooth receiver that makes it possible to display pictures coming from a Bluetooth device (PC, phone). You would like to send the photos of your phone to your family TV even though you are at the other end of the world. Of course, you want it to happen seamlessly as if your TV Bluetooth receiver were local. This is the kind of features that is supported by the UbiPAN system that we present in this paper.

To make the understanding of our approach easier, we need to fix some vocabulary. In the rest of this paper, a network composed of Bluetooth-enabled devices that are close to each other will be referred to as a *Bluetooth network*. Bluetooth networks include the piconets and the scatternets as defined by the Bluetooth Special Interest Group. In a Bluetooth network, devices thus interact without the mediation of any access point or repeater. We will refer to the extended network constructed by using our approach as a UbiPAN. A UbiPAN, as illustrated by Figure 1, is constructed by interconnecting several Bluetooth networks using gateways which will support interactions between remote devices. This will be described later in this paper.

Contributions. The worth noting contributions of this work are as follows:

- We propose a new approach to gracefully extend Personal



Figure 1. Overall architecture of UbiPAN

Area Networks by combining the use of wired and wireless communication technologies. We rely on the use of the IP network, the SIP [4] protocol and Bluetooth.

- We provide customized gateways to deploy a UbiPAN network that fulfills all the requirements of our approach.
- We conduct a test scenario to assess the system. The experiments, which consist in exchanging files, show that UbiPAN allows interactions between remote devices almost as seamlessly as in the case of interactions between devices of an ordinary Bluetooth network and with little overhead.

The rest of this paper is organized as follows. Section II presents the UbiPAN infrastructure. Section III describes how the Bluetooth service discovery and service delivery mechanisms are achieved in a UbiPAN. Section IV assesses our prototype. We show that the approach introduces little additional latency in data transmission. Finally, we discuss related work in Section V before drawing conclusions and sketching future work in Section VI.

II. UBIPAN INFRASTRUCTURE

UbiPAN relies on a network of distributed gateways to provide a transparent communication infrastructure to Bluetooth devices. These gateways are interconnected so as to exchange information about the Bluetooth services which are available in their own vicinity. Remote devices evolving in UbiPAN can then interact with each other *via* Bluetooth without being aware of the underlying connection infrastructure. As illustrated by Figure 1, gateways act as transparent bridges between distant Bluetooth networks.

Furthermore, UbiPAN has been designed with scalability in mind. Our approach leverages on the capabilities of IP-based networks and is centered around the notion of *presence* to overcome the issue of volatility of both the gateways and the Bluetooth devices/services.

Besides, we rely on a specific release of the UbiGate [5] gateway, to which three main features have been added: (i) UbiGate services have been made accessible on Bluetooth networks instead of only IP/SIP networks, (ii) we have added distribution capabilities to the UbiGate gateways, allowing

them to reliably share information relating to the evolution of their respective environments, (iii) we have provided them with Bluetooth service delivery capabilities to enable end-to-end Bluetooth communication between remote devices.

A. Constructing a UbiPAN network of gateways

In UbiPAN, UbiGate gateways are the backbone of the extended Bluetooth network: the tighter their interconnection, the more reliable UbiPAN. To contribute to the extended network, each UbiGate registers itself by UbiPAN as a representative of the Bluetooth network it belongs to. A registered gateway can then transfer all the information related to its own sub-network to the other registered gateways. Since these gateways mediate between the distant Bluetooth networks, real-time availability of the information about the gateways and the services they provide is required. To achieve this goal, we rely on the capabilities of the SIP¹ application level protocol [4] and its SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) extension for managing presence notification [6]².

In UbiPAN, a registrar server records the IP addresses of the available gateways (see Figure 1). When a gateway registers to take part in UbiPAN, it is automatically notified of the presence of the other UbiGate gateways available in the network. However, it must register by the other gateways to get informed of the Bluetooth services that are available in their vicinity (see below). By doing so we decentralize UbiPAN making the infrastructure more resistant to failures. In our approach, the centralized registrar server is only required when constructing the network, and the interactions between the different sub-networks in UbiPAN are directly managed by the gateways themselves.

B. UbiGate : Bluetooth over IP

The gateways deployed in UbiPAN are instances of the UbiGate gateway that was originally designed in previous

¹Session Initiation Protocol

²It can be noted that SIP, standardized by the IETF and adopted by the ITU (International Telecommunications Union), is originally a signaling protocol for Voice over IP (VoIP).

work to unify service discovery mechanisms [5]. This gateway copes with the volatility of services in pervasive environments by transforming discovery information into presence information. Practically, UbiGate offers an event based framework in charge of managing a SIP presence server. The architecture of UbiGate is based upon three major components called *managers*. A *Presence Manager* manages the SIP presence server, handles the storage of services as entries, and matches subscriptions for these entries. A *Communication Manager* handles the native discovery protocols supported by the gateway. A *Gateway Manager* strengthens the coordination between the components. It enables them to conjointly perform discovery and notification.

UbiGate was not originally designed to be deployed in an infrastructure network. We thus have designed and implemented an additional component, the *network infrastructure Manager*, to support the collaboration capacities required in this kind of environment.

We also took advantage of the extensibility capacity of UbiGate to propose a Bluetooth module that enables Bluetooth service delivery from the gateway to devices located in a Bluetooth network.

III. UBIPAN SERVICE DELIVERY PROCESS

In UbiPAN, a gateway is the *proxy* node that Bluetooth-enabled devices use to access resources of other sub-networks. The gateways are therefore the central nodes that manage Bluetooth service discovery as well as Bluetooth service delivery.

1) *Bluetooth Service Discovery*: Bluetooth service discovery in UbiPAN can be split in two distinct phases. In the first phase, information related to the services available in a sub-network are spread among the UbiPAN gateways. In the second phase, every gateway advertises the remote services that are available in its local Bluetooth network and that have been discovered.

Gateways rely on the Bluetooth Service Discovery Protocol (SDP) to discover and keep track of the services available in their local Bluetooth sub-network. Figure 3 (1) provides the sequence diagram of the messages exchanged to achieve a transparent service discovery in UbiPAN. The information that are recovered are relayed to the other gateways. However, because of the dynamicity of Bluetooth networks, we need to ensure that the broadcasted information is reliable and up to date. Therefore we use SIP to ensure that the interconnected gateways always have a recent snapshot of the context, which includes the services offered by the devices operating in UbiPAN.

In a Bluetooth network, the discovery process is carried out in two steps:

- a) The *Inquiry Process* [2], during which a device discovers the MAC addresses of the different Bluetooth-enabled devices in its vicinity.
- b) The *Service Discovery* that is performed by querying each device discovered in a).

The results of this discovery process is a list of UUIDs (Universal Unique Identifiers) that identify the available services that can be delivered by the devices in the vicinity. We have added to UbiGate the capacity of seamlessly advertising the list

```

1 user@host:$$ sdptool records --raw local
2 Sequence
3   Attribute 0x0000 - ServiceRecordHandle
4     UINT32 0x00010001
5   Attribute 0x0001 - ServiceClassIDList
6     Sequence
7       UUID16 0x1105 - OBEXObjectPush
8   Attribute 0x0004 - ProtocolDescriptorList
9     ...
10    ...
11   Attribute 0x1111
12     String 00:E0:03:60:4F:26

```

Figure 2. Extract of a Bluetooth service record in UbiPAN

of remote Bluetooth services based on their UUID (Universal Unique Identifier).

But the following problem remains. Each service in Bluetooth is identified by a UUID. Most devices use this UUID as a key to work with the services and to present them to the users. Thus, these devices are not able to present two services with the same UUID and one (usually selected at random) will be arbitrarily proposed to the user. The other service will thus remain unusable. It means that if mobile A and mobile B both offer a given service S (the Bluetooth file transfer service for instance), S will have the same UUID on both mobiles, and either S on A or S on B will remain unusable because it will never be presented to the final user. To resolve this issue, UbiGate gateways add, when creating the list of services that they offer, an attribute specifying the MAC address of the device that effectively offers the service (see Figure 2, lines 11-12) and this enables to make the difference between two instances of the same service.

2) *Bluetooth Service Delivery*: To control data transfer between devices, the closest gateways located in their respective vicinity establish a SIP communication session. As illustrated by Figure 3 (2), when a virtual service advertised by a gateway is invoked, a SIP session (SIP INVITE/OK/ACK - BYE/OK) is initiated with the remote gateway that has natively discovered the effective Bluetooth service. Data transfer can then be achieved (see Figure 3 (3)).

IV. ASSESSMENT

To assess the system, we have set up a simple file exchange scenario. In this scenario, we assume that Jane and her friend, Fred, are accustomed to exchange files using the Bluetooth wireless connection of their mobile phones. Fred, who has just flown overseas, would like to transfer a few souvenir pictures to Jane just as if they were close to each other. Let us assume that both Jane and Fred devices are within range of gateways that take part in the same UbiPAN network. Launching his file transfer application, Fred then discovers Jane's phone and can start an apparently classical Bluetooth file transfer, even though they are far from each other and thus not in the same Bluetooth network.

For the purpose of testing this scenario in UbiPAN, we have constructed two separate Bluetooth networks, each comprised of one single mobile Bluetooth equipment and hosting an instance of the extended UbiGate gateway (connected to

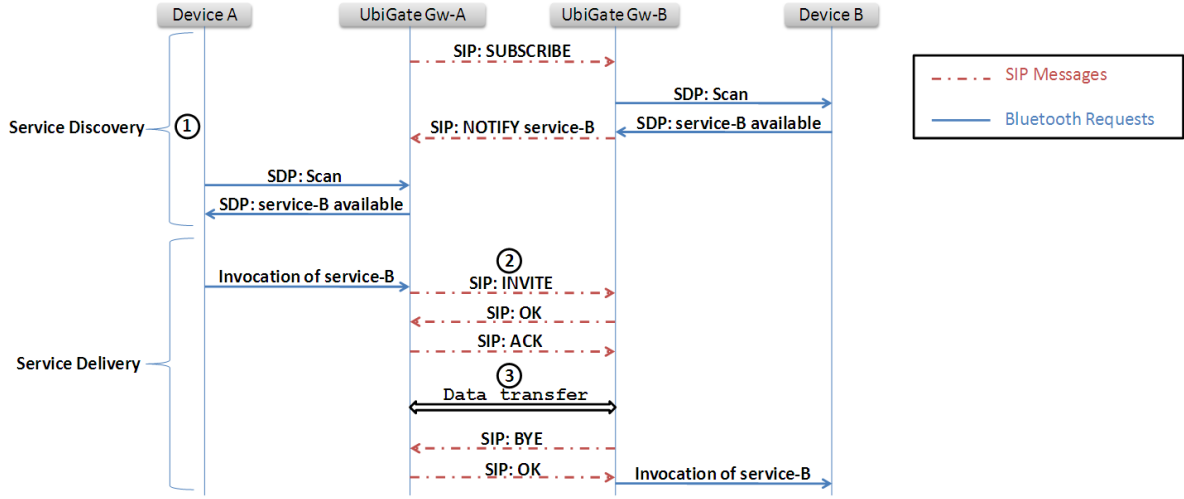


Figure 3. Sequence diagram of interactions between remote Bluetooth-enabled devices in UbiPAN

the other through a local Ethernet network) deployed on an Intel Pentium 4 computer with a 3 GHz CPU and 1 GB of memory. The transfer is achieved using the Bluetooth OBEX (OBject EXchange) profile. The extension of UbiGate to support this profile has required about 2,400 lines of Java code, bringing its total number of lines to 12,900. The new Bluetooth module in UbiGate, which provides Bluetooth service delivery capabilities, has a memory footprint of 48KB, including service discovery, and is based on the *AvetanaBluetooth* JSR82 implementation.

A. Evaluation of service discovery time

We have compared the UbiPAN Bluetooth discovery of local devices with the UbiPAN Bluetooth discovery of remote devices. We were able to confirm that our approach does not introduce any significant overhead between a local and a remote discovery. As shown in Table I, a gateway takes on average 31 seconds to complete the process - with the *AvetanaBluetooth* stack - in its local Bluetooth network. Then, after about 1 second, all its counterparts in the network can advertise the information in their respective local Bluetooth network. Our experiments consider only first inquiry sessions because, for subsequent discoveries, the devices use cached information from previous inquiries to accelerate the process.

Discovery type	Duration of the 1 st inquiry
UbiPAN local discovery	≈ 31.0 seconds
UbiPAN remote discovery	≈ 31.7 seconds
Overhead	+2.6%

Table I
COMPARISON OF BLUETOOTH INQUIRY DURATIONS

In the configuration of Figure 1, let G_A and G_B be the gateways deployed for our test, and let the mobile devices in their vicinity represent the Bluetooth-enabled devices willing to exchange the pictures. At the end of the discovery phase, G_A advertises the file transfer service of devices B , C and D . The transfer process is then initiated by device A in two steps: (i) the selection of the appropriate service and (ii) the

invocation of this service using its standard Bluetooth file transfer application. To allow the user of device A to identify the OBEX Object Push service that corresponds to device B , we have implemented a J2ME MIDlet that is installed on device A . The reason for having this MIDlet is as follows. Despite the fact that we have added an attribute to the services description (see Section III), typical file transfer applications and/or (mobile) operating systems set apart the services by comparing only their UUIDs. The J2ME MIDlet solves this problem by performing a classical *inquiry process* to discover any reachable gateway in the UbiPAN network infrastructure. Then, it performs a classical service discovery by inquiring the SDP server located on the gateway. This discovery phase yields, on the user's device, the list of remote services that can be accessed through this gateway. The MIDlet takes into account the attribute we have added (MAC address) and presents to the user two services with the same UUID as two different services. We believe that the possibility offered by our MIDlet to present several instances of the same service available on different devices will soon be build in commercial terminals. This MIDlet will then be useless and the use of UbiPAN will be even simpler (no need to install this MIDlet).

B. Evaluation of service delivery time

We now present preliminary performance evaluation results of (Bluetooth) service delivery between devices in a UbiPAN network. Table II provides the different running times necessary to transfer a 16 KB file.

Delivery type	Duration of the transfer
Classical Bluetooth delivery	≈ 12.57 seconds
UbiPAN remote service delivery	≈ 19.30 seconds
Overhead	+53.5%

Table II
COMPARISON OF SERVICE DELIVERY DURATIONS

The provider device is a *K750i Sony EricssonTM* mobile phone. The file transfer application of this device takes about 12 seconds, including the discovery steps, to transfer the file to

a computer using the *Bluez*³ Bluetooth stack on GNU/Linux. In UbiPAN, the same transfer from device *A* to device *B* through gateways G_A and G_B takes about 19 seconds, excluding previous discovery steps. For transferring data between the gateways (see Figure 3 (3)), we have used the Real-time Transport Protocol (RTP) [7].

The noticeable service delivery overhead stems from the fact that, in our prototype implementation, the transfer is not achieved as a zero-copy service. The file to transfer is copied from one entity to the other (mobile, gateway, gateway and then mobile), what introduces additional processing delays for initiating and terminating sessions. The conversion achieved by the underlying systems between Bluetooth and IP (packet size, etc.) is also likely to play an important role. This will be studied in more details and optimized in future work.

V. RELATED WORK

In recent years, various approaches have been proposed to try to generalize the use of Bluetooth as a primary communication technology. The industry has proposed access points and repeaters to widen the access range of Bluetooth-enabled devices. However, as opposed to the gateways in UbiPAN, these intermediate entities do not support graceful integration.

The Bluetooth specifications natively allow Bluetooth-enabled devices to constitute extended networks by interconnecting several piconets. These resulting scatternets [2], however, usually face difficulties in their operation because, at the edge of two piconets, frequency hop collisions can arise in the bridging device. Hu *et al* have designed novel algorithms to better regulate the frequency hops in scatternets [8] and try to solve the interconnection problem. Similarly, Chek *et al* propose in [9] management schemes to improve the coexistence of Bluetooth and WiFi in the same environment. Since both technologies share the same ISM (Industrial, Scientific, Medical) frequency spectrum, interferences are unavoidable and can lead to severe disrepairs in transmission performance. Nevertheless, even if it these proposals solve local connection problems, it is noteworthy that our approach of network extension is about addressing the issue of long range distance between remote devices, not local connection.

Finally, the Bluetooth Special Interest Group has recently released a new specification [10], Bluetooth High Speed v3.0, that will offer a huge increase in bandwidth (up to 24 Mbps instead of only 3 Mbps allowed by the v2.1 + EDR standard). To achieve this speed, the new standard borrows WiFi's 802.11 radio technology. Practically, the new chips will have the ability to wake up 802.11 radio on the device to transfer larger files and perform video streaming. Even if, with v3.0, high speed no longer requires a network infrastructure, our UbiPAN network infrastructure is still required (i) to discover remote services before any data transfer can occur, (ii) to deal with out of radio range devices.

VI. CONCLUSION AND FUTURE WORK

In this paper we have presented the design and evaluation of UbiPAN, a novel approach for constructing extended Bluetooth networks. A UbiPAN network infrastructure enables

users to interconnect their remote devices in the most seamless and effortless possible way as in a classical Bluetooth network even though they are far from each other. Using IP networks as well as the SIP and RTP protocols, UbiPAN reliably extends the range of PANs while dealing with the volatility of pervasive environments. Our experiments show that UbiPAN introduces an acceptable latency in data transfers compared to an ordinary Bluetooth network.

In terms of future work, several points will be dealt with. First, we will deploy UbiPAN in a real setting at our laboratory to assess it further (we have some 250 members and two buildings with three floors each). Second, we will study if the system that we have designed can help to support roaming. As of today, it is only possible for the terminals to move between each service request/delivery. Roaming should support the terminals moving during service delivery. Third, zero-copy service delivery should be supported. The goal is to have data delivered without any local copy being made at any point in the network. This should lead to better performance. Fourth, security related issues will also be considered.

REFERENCES

- [1] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, no. 3, pp. 66–75, September 1991.
- [2] "Bluetooth Special Interest Group SIG, Core Specification v2.1 + EDR," July 2007, <http://www.bluetooth.com/Bluetooth/Technology/Building/Specifications/>.
- [3] "IEEE 802.11 Specification," <http://standards.ieee.org/getieee802/802.11.html>.
- [4] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," Internet Engineering Task Force, RFC 3261, Jun. 2002, <http://www.rfc-editor.org/rfc/rfc3261.txt>.
- [5] T. Bissyandé, L. Réveillère, and Y.-D. Bromberg, "UbiGate: A Gateway to transform Discovery Information into Presence Information," in *SIPE'09: 4th international Workshop on Services Integration in Pervasive Environments*, London, UK, July 2009.
- [6] J. Rosenberg, "A Presence Event Package for the Session Initiation Protocol (SIP)," Internet Engineering Task Force, RFC 3856, Aug. 2004, <http://www.rfc-editor.org/rfc/rfc3856.txt>.
- [7] A.-V. T. W. Group, H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force, RFC 1889, Jan. 1996, <http://www.rfc-editor.org/rfc/rfc1889.txt>.
- [8] C. W. Yu, K.-M. Yu, and S. L. Lin, "Efficient scheduling algorithms for bluetooth scatternets," *Wirel. Pers. Commun.*, vol. 48, no. 2, pp. 291–309, 2009.
- [9] M. C.-H. Chek and Y.-K. Kwok, "Design and evaluation of practical coexistence management schemes for bluetooth and ieee 802.11b systems," *Comput. Netw.*, vol. 51, no. 8, pp. 2086–2103, 2007.
- [10] "Bluetooth Special Interest Group SIG, Core Specification v3.0 + HS," April 2009, http://www.bluetooth.com/Bluetooth/Technology/Works/Core_Specification_v30.htm.

³<http://www.bluez.org/>