The Broadband Phone Network: Experiences with Context-Aware Telephony
$CUED/F - INFENG/TR.533$

Ripduman Sohan, Alastair Tse and Anil Madhavapeddy
Laboratory For Communications Engineering
{rss39,acnt2,avsm2}@cam.ac.uk

March 2005

Abstract

   We introduce the Broadband Phone, a Voice-over-IP (VoIP) based phone which integrates a high-resolution, touch-sensitive LCD screen. The Broadband Phone Network features: (*i*) a thin-client network architecture enabling easy integration into the *sentient environment* at the Anonymous University; (*ii*) location-sensing via the Active Bat indoor location system allowing applications such as the "follow-me phone"; and (*iii*) location-based handover of Bluetooth connections, enabling integration of the users' own personal mobile phones. We describe the thin-client network architecture, the applications which use it such as the new dialing mechanisms, and also how the Broadband Phone can be used as a peripheral display. Finally, we relate our experiences with more than a year of operating this phone network and discuss the pros and cons of our approach.

1   Introduction

The modern knowledge worker carries around multiple personal devices (e.g. mobile phones, PDAs and cameras) to help them in their day to day activities. These devices are often difficult to integrate with each other and typically lack *context-awareness* [8] — users must manually switch their phones to "quiet mode" in a meeting for example.

   We investigate an alternative to the trend of giving users "heavy clients" that encapsulate logic and storage in a single device. Rather, we propose using the *sentient environment* [1] — a location which consists of sensors and wireless networks that enable users to roam while carrying only a single, lightweight location sensor. Services traditionally provided by user's personal devices are now available through displays and interfaces spread through the environment.

   Our implementation of a sentient environment at the Anonymous University[1] consists of the following components: (*i*) the Active Bat indoor location system [8] which provides real-time location information to applications; (*ii*) the Broadband Phones (BBphones), which are telephones with an integrated, touch-sensitive LCD screen and Voice over IP (VoIP) capability; and (*iii*) the *Bluebone* network of Bluetooth nodes which remain connected to users' mobile devices using location-based handover.

   This paper describes the thin-client architecture behind the Broadband Phone Network (BBnet), explains how the other portions of the sentient environment are integrated into it, and outlines the novel applications that resulted from the work. We also discuss the pros and cons of our approach after a year of experience operating the network with 40 handsets at the Anonymous University.

   Our thin-client approach has the following advantages: (*i*) users only have to carry a single, lightweight location sensor with them to benefit from all the services that the environment provides; (*ii*) the environment can pool contextual information from multiple users, which is much harder to do if they are all using

---

[1]For anonymous review, we refer to two institutions as the "Anonymous University" and "Anonymous Laboratories", and figures are blurred to hide logos and names.

2

Figure 1: The Broadband Phone

different and incompatible personal devices (a good example is a group meeting automatically rerouting all non-critical calls to voicemail); and (*iii*) new services can be rolled out to users by upgrading software centrally in the environment since users do not carry a lot of "mobile state".

Our thin-client approach provides an excellent research testbed for rapidly prototyping and deploying technologies which would substantially take more time and effort with a conventional heavy-client approach. We have demonstrated that such an approach is feasible for a building-wide deployment.

1.1 The Broadband Phone

The Broadband Phone (BBphone) is a custom designed hardware device based on the StrongARM 1100 processor, containing 8MB of flash memory and 32MB of RAM. The devices is equipped with a Cirrus Logic 10 Mbps Ethernet network interface, a customised sound-card derived from the Cirrus Logic CS4235, a touch-sensitive LCD display, and circuitry for handset and speaker-phone functionality. The phone runs an operating system based on the Linux 2.2 kernel (modified with custom drivers and scheduling logic). The phone is covered in a molded hard-plastic casing designed to give the device the "look-and-feel" of a traditional telephone handset (see Figure 1). Users can interact with the touch-screen with a stylus for more accurate input.

The BBphone performs very little computation and acts as a thin-client (see Section 2.1); the screen is driven remotely using the Virtual Network Computing (VNC) [21] protocol and voice calls operate via standard VoIP. Calls are signaled using the Session Initiation Protocol (SIP) [22] and transported using the Real-time Transport Protocol (RTP) [24].

1.2 The Active Bat System

The Active Bat system is a sophisticated indoor location system in which small active "Bats" are signaled periodically to emit narrowband ultrasound pulses.

These pulses are detected by sensors in the ceiling, and the time-of-flight information is used to multi-laterate the position of the Bats - a process which is accurate to 3cm 95% of the time.

The Active Bat system can trigger 50 location updates per second per radio zone; there are two such zones in the current installation. The system adaptively schedules Bat updates and offers highly mobile Bats greater accuracy by scheduling them more often. Applications can also specifically request a higher update rate from the schedule for one or more Bats for limited periods of time.

The Active Bat system provides higher update rates and level of accuracy than are likely to be found in any deployed commercial location system in the near future. It provides an ideal research platform for determining the features *future* location systems must have to support applications such as context-aware telephony.

Instead of forcing applications to process a high volume of raw location updates, the Active Bat system provides the SPIRIT [1] spatial indexing middleware. SPIRIT represents location and static resource data (the *world model*) by a series of persistent CORBA objects using the omniORB [11] system. Around 40 different object types are registered, such as "person", "camera", "printer", and "phone", and a quadtree based indexing method is used to quickly determine space containment and overlap relationships.

Quiet zones are special areas which users enter to indicate they do not wish to be disturbed. These quiet zones can be created by the user using their Bats or dynamically imposed by the environment (e.g. when a meeting is in progress).

Spatial buttons are regions of space which have some action associated with them. The space is generally marked by a poster, and the user clicks their Bat in the space to perform an action. These spatial buttons are simple to create and position, easy for users to activate, and provide users with an incentive to wear their Bats on a regular basis [14]. Examples of spatial buttons deployed around the environment are: (i) users waiting for a colleague to return to their office click on a poster outside the office and are informed when they return; (ii) users can create a quiet zone when they do not want to be disturbed, and (iii) users can be notified when fresh coffee is available.

2   The Broadband Phone Network

2.1   Thin-Client Systems

Thin-client architectures, defined as those systems where the bulk of the processing is done on the server, are becoming increasingly prevalent [10, 21, 20]. A typical thin-client system consists of three parts: (i) the client acting as an input/output device, (ii) the server acting as a processing unit, and (iii) a fast communication network linking the two.

The most prevalent thin-client system in our time is the telephone network. The handset only requires an electrical connection to the network and is able to signal calls and modulate voice with little complexity. Call routing

and frequency-division multiplexing are performed by the telephone exchanges. While the exchanges have seen orders of magnitude of increase in data transfer rates over the last century, the handsets have only required a single upgrade from pulse based to tone-based dialing. In contrast, modern mobile phone networks are increasingly moving away from the thin-client model. We observe that mobile phone handsets require regular replacement to enjoy new services offered by the network due to much of the intelligence being put within the device itself, rather than inside the network.

Several advantages are offered by the thin-client computation model. The centralised nature of the system allows for simpler maintenance, security and management. Client hardware complexity and cost can be greatly reduced in this model. New services can also be deployed rapidly as there is little or no requirement for users to update their client hardware.

Traditionally, the large-scale deployment of the thin-client model has been impeded by its substantial technical requirements [32, 23]. Thin-client systems require networks with low latency and sufficient bandwidth to accommodate the input/output of the clients. Since the server is performing the bulk of processing, it requires fast processing power and memory. These requirements were difficult to achieve for large systems, and hence past work focused on providing single-user remote access to workstation desktops and applications. However, recent advances in cluster computing [31] and server virtualisation [2, 27] have made it technically feasible to construct *thin-client servers*, software systems designed to simultaneously service a number of collaborating clients.

Our BBphone network combines portions of the traditional phone network and modern VoIP systems by adopting a thin-client approach that places the minimum required functionality for VoIP functionality and remote display in the handset, leaving any complex processing such as call routing to central servers.

2.2   Network Architecture

Figure 2 illustrates the overall architecture of the Broadband Phone Network (BBnet). The BBphone touch-sensitive LCD screen is driven by the VNC protocol. VNC provides a single graphics primitive that places a rectangle of pixel data at a location *(x,y)*. Interaction occurs in a client-server manner; the phone endpoint is known as the *VNC client* and the server where the display changes occur is the *VNC server*. User input events are transmitted from the VNC client to the VNC server.

The BBphone VNC client polls the server for frame-buffer updates at a fixed rate. The client can either request a full screen update, or incremental changes from the previous update. Various update encoding schemes allow for a large degree of flexibility in trading parameters such as bandwidth and server CPU.

The Session Initiation Protocol (SIP) [22] is a signaling protocol used for establishing sessions in an IP network. A session is typically a two-way VoIP call, but can also be a complex, multi-party multimedia conference call.

When a user places a call to another BBphone, the BBnet server receives the recipient details from the phone via VNC input events. A SIP INVITE message
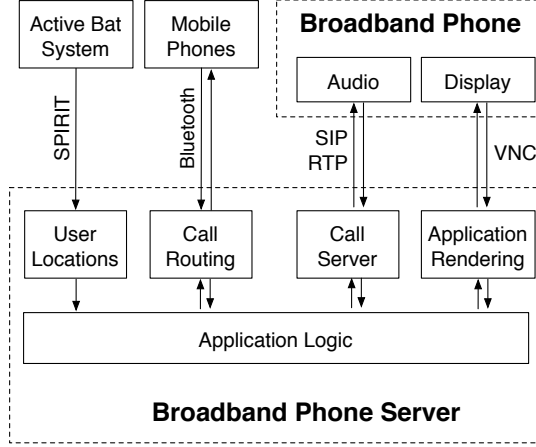
Figure 2: The overall architecture of the Broadband Phone Network (BBnet)

is then sent to the recipient BBphone. Phones which are free to receive calls will issue a ring tone, while busy phones reply with SIP DECLINE messages. The BBnet server will issue SIP ACK messages to all participating phones which should open voice connections to each other.

The phones send voice data to each other in three stages: (i) raw sound data is read from the handset; (ii) the data is transcoded to a specified audio codec[2]; and (iii) an RTP packet is created and transmitted to the recipient phones. In our system, the phones only act as simple audio-to-IP devices, leaving the BBnet server to perform all the call routing and establishment. In an effort to minimize call latency, the thin-client architecture is not extended to routing the RTP voice packets via the BBnet server, instead sending them directly between phones in a peer-to-peer fashion.

The BBnet server uses the TiCL thin-client architecture [26] developed as part of this work. TiCL is based on the principle of independent modules which communicate via UNIX sockets (e.g. domain sockets or TCP). Each module has one or more specialised task related to driving the BBphones or gathering data from the sentient environment (e.g. location information from the Active Bats). TiCL's modular architecture allows services to be split across multiple machines, allowing for easy scaling and redundancy with commodity PC hardware.

Every connected BBphone client spawns an instance of a special TiCL *Session module* which encapsulates that phone's state for the lifetime of the client. This session includes instances of the *Application Renderer* and *Application Logic* modules. The Application Renderer runs the VNC server to display graphics on the BBphone display, and uses a modified version of Trolltech's QTopia windowing toolkit [9]. The Application Logic module is the repository of programs (written in a combination of C++ and Python) which parse in-

---

[2]The codec is negotiated as part of the initial SIP invitation

put events, accept information feeds from the environment, and instruct the Application Renderer module to display the appropriate widgets.

The BBnet server also spawns several global TiCL modules. The *Call Server* listens for call requests from phones and acts as a SIP proxy (e.g. it issues SIP invitations to recipients). The *User Locations* module accepts input from the SPIRIT spatial indexing middleware, giving the BBnet server updates on the location of users in the building (see Section 3.1). The *Call Routing* module integrates users' Bluetooth mobile phones into the BBnet, by allowing incoming and outgoing calls to be routed via the GSM mobile network (see Section 3.2).

3 Integration with the Sentient Environment

Sentient computing [1] is a method of realising the ubiquitous computing vision using sensors in the environment to allow devices and applications to react accordingly. A common use of the sensors is to create a "world model" which allows location-aware or context-aware applications to be constructed.

There are two major sensor networks in the current deployment: (*i*) the Active Bat indoor location system provides applications with accurate user location information; and (*ii*) the Bluebone is a network of Bluetooth nodes distributed through the building to interface with user's personal mobile devices. This section describes how the Broadband Phone network integrates with these two technologies to provide context-aware telephony services.

3.1 Location-Aware Telephony

In order to provide location-aware services, the BBnet server uses SPIRIT to obtain two levels of information: (*i*) a room-grained feed which informs it as users leave and enter rooms; and (*ii*) notification when a user clicks their Bat above a particular BBphone.

There is a simple decision process for routing calls appropriately. Users nominate a "secondary recipient" for calls to be sent to if they are not available — this is most often their voicemail, but can also be a secretary or just a simple vacation message. The BBnet server looks up the user's current location, and infers whether the user is able to receive calls by: (*i*) finding a free BBphone in range of the user (there are no phones in the bathrooms!) (*ii*) examining the location to see if it is a "quiet zone" (e.g. a meeting); and (*iii*) checking the user is not already in a phone call.

Meetings are started by clicking a spatial button on a poster in the room. By default, they last for an hour and are established as quiet zones (if they finish earlier, they can be terminated by using another spatial button). Any other users entering the room during this time are assumed to be part of the meeting and participate in the quiet zone (i.e. they do not need to manually set their status). Although more automated methods of "activity inference" to determine the status of users can also be used [18], in our deployment we prefer the more explicit user-driven approach to avoid frustrating lapses.

More expressive policies for call routing are also certainly possible. Cadiz et al. observed in their extensive Enhanced Telephony user study [3] that of all the features of their prototype (call dialing, in-call options and incoming call routing), users rated the incoming call routing flexibility as the most useful feature. We note that incoming call routing is analogous to email filtering, and are currently investigating exposing this feature to users in a manner similar to `procmail` [29]. Users can then write simple "recipes" (a set of regular expression matches on the incoming call information and their location) to cater for their own unique situations (e.g. allow a critical call to penetrate quiet zones, or block calls during morning hours). Another option possible with the BBphones is to "push back" context to the caller by notifying them that the target is currently busy in a meeting, and give the caller the option to continue with the call. Currently, the context transmitted back consists of the names all of the members in the quiet zone — we have not addressed the privacy concerns that may arise as a result of this in deployments where users do not trust each other.

Recall that the BBnet is also informed when users click their Bat above a BBphone. This is used to let users control the phone display to form a *lightweight session* that is automatically logged out when they move out of the immediate vicinity of the phone. BBphones are spread throughout the environment and spend most of their time idle. Contrast this to desktop PCs, which are often left logged in for long periods of times and are generally assigned to a single user, making them difficult to take over for a quick query. When the user takes control of a BBphone, its status bar displays the user's name, and applications are loaded with that user's data. Applications which make use of this functionality are discussed in Section 4.1.

3.2 Accessing Mobile Phones using the Bluebone

One key problem in operating a VoIP phone network is determining an appropriate charging model. Since Anonymous University only provides external and internal telephone connectivity via the traditional telephone network, expensive PBX-to-VoIP conversion hardware is required to integrate the BBnet with the outside telephone network.

However, over the past few years, an increasing number of mobile phones on the market support the Bluetooth wireless protocol. Bluetooth is a short-range, wireless "cable-replacement" protocol operating in the unlicensed 2.4GHz spectrum. Popular uses for Bluetooth in mobile phones include media transfer (e.g. pictures and movies), multiplayer gaming and wireless headsets and speakers.

Integrating these mobile phones with the rest of the sentient environment solved a number of problems: (*i*) users were charged for their individual phone usage via their normal mobile phone billing provider; (*ii*) users' incoming mobile calls are automatically routed to the BBnet when they are present in the Anonymous University; and (*iii*) external network services such as Caller Line Identification (CLI) worked much better as calls originated from the user's phone rather than an office exchange. In return for the integration, users benefited from not having to carry their mobile phone with them while at work (a com-

mon source of irritation is forgotten mobile phones ringing loudly until their owners run back to answer them), and also not having to keep its status in sync with their current mode of work (e.g. forgetting to activate the "silent" profile in a meeting).

Bluetooth provides two connection mechanisms over its base-band radio layer; (i) the Logical Link Control and Adaptation Protocol (L2CAP) which provides a reliable protocol layer with features such as multiplexing, segmentation and reassembly, service groups, and quality of service support; and (ii) the Synchronous Connection-Oriented (SCO) protocol which provides support for real-time voice traffic via reserved bandwidth. The RFCOMM protocol provides an emulation of serial ports over the L2CAP layer, providing simple flow-control and "ports" applications can connect to.

The Bluetooth Special Interest Group [5] defines a series of "Bluetooth Profiles" which define use-cases for specific application domains. The "Handsfree Profile", designed for use by handsfree kits commonly seen in automobiles, provides the complete set of functionality required to integrate the mobile phone with the BBnet. This profile provides a Bluetooth interface to transmit voice data bidirectionally via SCO, and uses RFCOMM to establish a control interface from the headset to the phone to dial external calls or accept or reject incoming calls.

The *Bluebone* network of Bluetooth nodes is used to interface with users' mobile phones, Most of the nodes are desktop PCs fitted with off-the-shelf USB Bluetooth dongles. Users pair their mobile phones to these nodes (which pretend to be Bluetooth headsets) as a one-off registration process. Bluetooth does not have support for connection handover between multiple nodes — since no single Bluetooth node can cover the entire building[3], a system for *location-based handover* was designed by using the Active Bats.

A Bat attached to a mobile phone was used to survey the building to gain coverage data for each of these nodes (see Figure 3). The survey typically took around 30 minutes for each Bluebone node, and resulted in around 40,000 samples of signal strengths and their locations. The survey was required as indoor signal reflection effects make it impossible to use standard Bluetooth propagation models to determine range accurately. A *spatial zone* is fitted to each node, which represents the area in which that node can provide good Bluetooth coverage.

When a user enters a spatial zone, an RFCOMM connection is made to the user's mobile phone (recall that the user has registered their Bluetooth phone address in the initial pairing) and handsfree control is obtained by the BBnet server. When the user moves out of range as reported by SPIRIT, the Bluetooth connection is terminated by the node in order to allow the node in the new zone to take over (see Figure 4). There is a period of a few seconds in each handover when no Bluebone node has handsfree control — this is impossible to avoid in the current generation of mobile phone Bluetooth stacks since they only reliably support a single RFCOMM connection at a time. However, if a phone call occurs

_____

[3]Bluetooth is a short-range wireless protocol, with typical indoor range of 8-10 metres
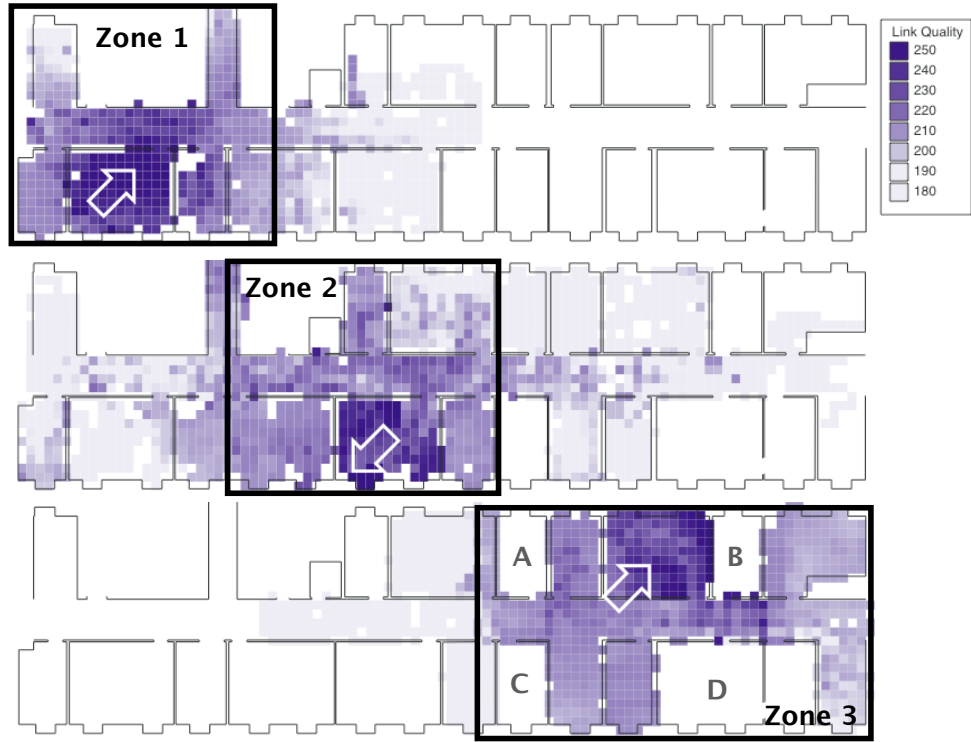
Figure 3: Three Bluebone nodes are placed at different locations. The figure shows three separate coverage surveys for the same floor. Three spatial zones are overlaid to illustrate where usable coverage exists (based on link quality values between 200 and the maximum 255). Four unsurveyed offices are marked rooms A to D.
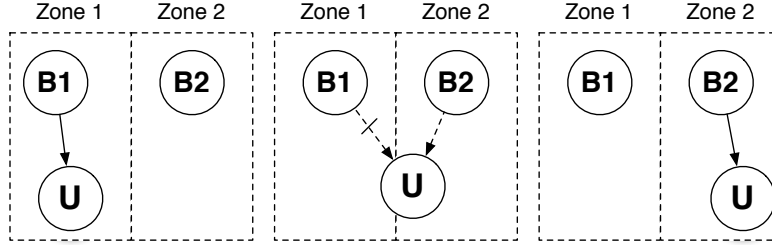
Figure 4: An illustration of location-based Bluetooth handover. Nodes B1/B2 represent Bluetooth servers in the environment, and U is the mobile user wearing a Bat. When the user is in the Bluetooth zone assigned to node B1, it establishes a connection to U's mobile phone *(left)*. When the user moves into another zone *(middle)*, B1 terminates its connection, and B2 establishes a connection *(right)*. The user is never mobile during a BBphone call, so the temporary interruption during the connection handover is acceptable as it does not interrupt an ongoing call.

during this time, the user's mobile phone rings as normal, so no calls are ever dropped or misrouted.

Note that this "location-based handover" made possible by the Active Bat system is better than the alternative of having every Bluetooth node perform continuous Device Discovery, as excessive Bluetooth traffic has been shown to interfere with any co-located 802.11b networks [7]. Since the building does operate a popular WiFi network as well, minimal disruption from the Bluetooth network was essential for user acceptance.

Since current mobile phones do not have accurate indoor location information available, location-based handover depends on the mobile phone being carried by a person wearing a Bat. However, it must also cope with the situation where a user leaves their mobile phone in their office. This is handled by the handover logic — when a user leaves the spatial zone, the Bluebone node checks the link quality to ensure that it is below a certain threshold (determined by the survey process). If the link quality is higher than the threshold, then this indicates the phone is actually still present in the zone, and the connection is maintained and further zone transition checks are disabled until the user re-enters that zone.

The Bluebone nodes acts as SIP clients to the Call Server to route traffic to and from mobile phones. When a user places a BBphone call to an external number, it is routed by the Call Server to their mobile phone (a user has to click their Bat above a BBphone to identify themselves before making such a call). The Call Server transcodes the Bluetooth audio codec format[4] into the BBphone codec in real-time and generates RTP packets to send to the BBphones.

---

[4] Although Continuously Variable Slope Delta (CVSD) modulation is used over-the-air, the Cambridge Silicon Radio Bluetooth dongles we used accepted the simpler Pulse Code Modulation (PCM) and performed the translation on the chip.

Figure 5: Face-based dialing from the BBphone *(left)* and a close-up screenshot *(right)*

4 Applications

The inclusion of a touch-sensitive LCD screen in the BBphone enabled the creation of a number of useful applications. The phone "applications menu" includes a calculator, world-time display, currency converter, and other useful utilities. The availability of location information enables other novel uses described below.

Since the BBphone has access to much more information about local users than a traditional telephone exchange, it offers some new dialing mechanisms in addition to the numeric keypads that are present in standard telephones.

Face-based dialing retrieves users' names and pictures from SPIRIT (see Figure 5). It then presents this information in a manner analogous to current Instant Messaging clients such as ICQ or AIM. The "contacts list" consists of a person's picture, their name, and their current location. If a user is unavailable (for example, in a quiet zone or out of the building), the picture is grayed out. Since this display doesn't scale well to a large number of users, users can speak their target's first or last name into the handset. The BBnet server then uses the CMU Sphinx [30] speech recognition framework to measure a set of matches ordered by confidence level. The displayed faces are "culled" by eliminating the bottom 75% of matches, allowing the user to quickly scan through the remaining matches and locate their desired user. This approach avoids the need to make an accurate voice match; by instead eliminating the worst matches, there is less need for users to "train" the voice recognition system.

Location-based dialing retrieves the locations of all the active BBphones from SPIRIT and overlays them onto a map of the building. The color of the phone icon on the map indicates its status: red for busy (engaged), green for available, and gray if the phone is switched off or otherwise unavailable. Users simply touch the location they wish to contact, and a call is placed. This mode is particularly useful for contacting "anonymous phones", such as the BBphones deployed in meeting rooms which have no fixed owner. To place a call into a meeting room or collaboration area, the user also doesn't need to look up

Figure 6: Two users collaborating using the BBphone "Scribblepad" application during a meeting. They can click their Bats above the phone to email the image to them.

numbers; knowing the location they wish to contact is sufficient.

Finally, *service-based dialing* provides a set of abstract services such as the "helpdesk" or "nearest system administrator". The Service Dialing application displays an HTML form to users with information to fill in before the call is placed (e.g. a Windows/Linux/MacOS checkbox for the system administrator service), and uses this information to route the call to the nearest appropriate person. This approach is similar to the Interactive Voice Response systems, but exploits the extra input capabilities provided by the touch-sensitive LCD to avoid forcing users to navigate cumbersome voice menus. In addition, the BBphone automatically submits the identity of the person dialing, which allows a degree of personalization in the call routing. Consider a helpdesk service which could attempt to assign the same staff to a user across multiple calls to avoid the user having to restate their problem repeatedly.

BBphones are deployed in collaboration spaces such as meeting rooms in addition to offices. Applications such as the "Scribblepad" (a free-form drawing and diagramming application) can be used by users as an electronic pad to exchange ideas during meetings (see Figure 6). Users can click their Active Bats above the BBphone in order to have the contents of the scribblepad emailed to them for future reference.

These applications are particularly useful due to the *lightweight login sessions* that the BBphones offer. Unlike desktop PCs which tend to have long-running sessions running on them owned by a particular user, the BBphones are designed to be used for short periods of time by many users — it takes only the click of a Bat to personalize a BBphone display, and it is automatically released when the user moves out of range. Similar functionality could be obtained by users carrying PDAs on their person as they move around. However, our approach of providing these services as part of the local environment minimizes the number of devices users need to carry with them — carrying a Bat even allows users to leave their cellphones on their desks as the Bluebone routes calls to the nearest BBphone.

4.1 Peripheral Displays

Peripheral (or ambient) displays are a class of output devices which operate on the periphery of human perception [13]. They require less specific attention from human users, instead relying on feeding impressions to viewers if they give it a quick glance in passing.

In a typical office building, personal computers are commonplace and have their own monitors. These computers are difficult to deploy as effective peripheral displays since the screen space is generally in use by the computer's owner. In contrast, the BBphones deployed at the Anonymous University spend most of their time idle, and are also deployed in locations where computers are not present (e.g. meeting rooms or comfortable collaboration areas).

The BBnet server detects when a phone has gone through a period of inactivity and launches a special *peripheral screensaver* application which displays various data sources using large, bright icons that can be noticed from a distance. These applications range from a weather station (which uses real-time data from a local weather station) to relevant location system events around the Anonymous University.

The "killer application" of peripheral displays comes from using the BBphone as a display to the Active Bat location system since the Bats lack sophisticated input/output capabilities [15] (they are limited to a buzzer and two buttons). When location-aware applications were initially developed, the buzzer was used to play simple tunes, and different tones distinguished messages from the location system. Some users complained that their Bats were now too noisy and that messages were hard to distinguish. The applications were modified to signal the user with a simpler beeps to the Bat when their attention was required. The user then clicks their Bat above the nearest BBphone to display additional information about the beep.

The *Virtual Receptionist* assists visitors to the Anonymous University who need to contact people inside the building (required to gain entrance). However, since members of staff and students are often away from their desks collaborating or meetings [5], they can be difficult for the visitor to locate. The Virtual Receptionist is a touch-sensitive LCD panel outside the building's front doors which displays the full list of staff, along with pictures of their faces and their current availability: (*i*) present in the building, (*ii*) away from the building, or (*iii*) busy (e.g. in a quiet zone). Visitors touch the icon of the person they wish to contact, and are requested to pose for a picture by a web-cam above the display. The contacted member of staff is alerted by a Bat beep, and can view the picture on their nearest BBphone. If they choose to accept the request, the LCD panel displays a map of the building and the staff member's location as they walk to the front door to greet the visitor.

The Bat system can be used to establish callbacks when a specified location event occurs, known as the *Alert Me* service. Examples of events include a colleague returning to their office, a meeting starting, or the coffee machine finishing a brew. When one of these events occurs, a beep is sent to the requesting

---

[5]or, more commonly, hanging out and drinking coffee!

user's Bat. When the Bat is clicked on a BBphone, the exact event is reported.

Another useful variation of the *Alert Me* application is to set the user's BBphone to flash automatically with the identity of anyone entering or leaving the building. By glancing at the BBphone display, they can know about events happening outside their immediate observable environment. Relevant impromptu gatherings and meetings can be easily spotted by the user purely by the activity increase on their peripheral displays.

5 Discussion

The Broadband Phones were originally designed in 1998 and initially deployed at Anonymous Laboratories. They were redeployed in 2003 with a rewritten software backend at the Anonymous University until present day.

The hardware behind the Broadband Phones has aged remarkably well over the last six years. However, operating the phones as thin-clients has ensured that the BBphone hardware requirements do not increase as the network size increases or more services are deployed. In contrast, the BBnet servers have been upgraded several times over the last year to cope with increasing demands such as the integration with mobile phone hardware and the Active Bat location system.

During the current deployment, user feedback of the system has motivated many of the applications described in this paper. The network faced similar problems to the Active Bat deployment [14], with an initial surge of interest when first deployed and usage subsequently tailing off. The addition of the "weather station" peripheral display and the subsequent Active Bat applications to locate other users in the building were popular.

A major problem was the inability to place and receive external telephone calls on the system after it was redeployed in 2003, due to the lack of an appropriate PBX. This resulted in the Bluetooth integration allowing BBphone calls to be routed directly onto cellular networks. This integration has proven challenging due to the immaturity of the various Bluetooth stacks and hardware dongles. Many popular phones (such as the Nokia Series 60) exhibit bugs in their SCO connections that have to be worked around, and only recent Linux kernel versions (2.4.26+) reliably support SCO connections. Other popular operating systems such as FreeBSD and MacOS X do not currently support isochronous SCO connections at all. On the USB dongle front, most are specialized for use in desktops or laptop PCs instead of servers, and hence only support a single SCO connection.

Research networks are typically run in an ad-hoc fashion, with frequent breakage and service unavailability due to experimental software changes. The BBnet avoided this situation due to the ease of administration and monitoring that results from the thin-client architecture. Administrators of the network can reboot phones remotely, and "share their display" by connecting to the VNC session from their own computers to investigate any problems.

The integration of the Active Bat infrastructure required some user training.

Since the BBphones are used as "spatial buttons" and registered with SPIRIT, users are not allowed to move them beyond rotating them on the spot. Moving the phones without informing the administrators resulted in inaccuracies in the "follow-me phone" service. However, this can easily be solved by an application that allows the user to reposition the phone in the world model by using their own Bat when they move the phone.

6   Related Work

The Broadband Phones are inspired by the pioneering Etherphone [33] developed at PARC through the eighties. Swinehart et al. created 50 Etherphone devices and deployed them around their laboratory. Users registered their locations by logging in a phone, and in return got personalized ring-tones, and could store and manipulate voice messages [28]. The BBphone project combines these ideas with a modern sentient computing environment, with location-sensing eliminating the need for users to "log-in", and the thin-client display technology providing a more sophisticated input/output for the user to interact with the environment.

We adopted guidelines for our use of Broadband Phones as peripheral displays from Project Aura [6], which focuses on minimizing the user's attention span in a pervasive computing environment. When experimenting with uses of the BBphone loudspeaker, Audio Aura [16] provided ideas for audible "sonic ecologies" although these were never deployed into the live BBnet.

Context sensitivity in telephony has also been considered by researchers in the realm of mobile phones — Project Sensay [25] augments a mobile phone with sensors such as accelerometers and ambient microphones to make contextual decisions such as the state of the ringer. In contrast, the BBnet integrates the user's existing Bluetooth mobile phone as part of the sentient environment, taking advantage of external sensors such as the Active Bats to provide the extra information needed to perform tasks such as indoor location-based hand-over or call routing.

In the commercial marketplace, VoIP phones are being sold by major vendors such as Cisco Systems [4], Nortel Networks [19] and Lucent Technologies [12]. The Cisco 7900 Series IP Phones [4] have simple LCD displays which can display XML-aggregated data. However, unlike the BBphone, arbitrary applications cannot be loaded onto the phones and the screen is not touch-sensitive. These extra features allow the BBnet to integrate with indoor location systems and Bluetooth mobile phones and provide a rich set of location-aware applications to users.

Cadiz et al. explored PC/Telephone convergence by installing applications on over 7000 PCs which communicated with a customized PBX [3]. They reported an "overwhelmingly positive" user reaction to their enhanced telephony prototype, with over 94% of the initial users recommending the system to their colleagues. Their system was a software-only solution (installed on user desktops), but highlights the value of context-awareness in telephone communica-

tions. The Broadband Phone system builds on this research by reporting our experience with deploying custom hardware solutions which closely integrate with the rest of a sentient environment, as well as integrating the users' mobile phones.

The Quiet Calls project [17] proposes alternative mechanisms for users to respond to incoming mobile phone calls, for instance when they are in meetings. They give users the option of listening to a call, and responding silently by using a set of menus on their phone. The Broadband Phone network also fulfills a similar role for users in our environment by routing mobile phone calls to a user's nearest phone and respecting that zone's access policy (e.g. if it has been designated as a quiet zone). The BBphone can then use its peripheral display functionality to flash messages to the user in the background, allowing them to respond if they consider the call important. Unlike the Quiet Calls solution, users are not limited to responding with the limited mobile phone input/output, but can use their local BBphone's touch-sensitive display to select responses.

7 Conclusions

We have described the concepts and architecture behind the Broadband Phone Network, as well our experiences at operating the network in a live environment with 40 handsets for over a year. The BBnet's thin-client architecture resulted in an extremely effective pervasive computing testbed due to the rapidity with which changes could be prototyped, implemented and deployed from a central location to remote BBphones throughout the building. This was demonstrated by integrating it with: (i) the Active Bat indoor location system, providing location-aware telephony services such as the "follow-me phone"; and (ii) the Bluebone network of Bluetooth nodes which enabled the integration of users' own personal devices such as their mobile phones. The work on the Bluebone resulted in the creation of a new location-based Bluetooth connection handover technique.

After this integration, the BBphones proved useful as local interfaces to location-aware applications, helping to overcome the limited input/output provided by the Active Bat system. In addition, new telephony applications such as the "follow-me phone" and dialing mechanisms (face-, location- and service-based) also became possible.

In the future, we intend to build upon this architecture by augmenting telephone conversations (which has not been a focus of our work so far). The BBnet infrastructure makes it straightforward to display shared whiteboards, contextual displays, live teleconference displays and other modern innovations from the field of Computer Supported Collaborative Work.

The Broadband Phones also provide us with dedicated displays dotted around the environment. The further integration of these displays with other services such as the location system and more advanced activity inference is an area which needs further exploration and user studies.

References

[1] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementing a sentient computing system. *IEEE Computer*, 34(8):50–56, August 2001.

[2] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *Proceedings of the nineteenth ACM symposium on Operating systems principles*, pages 164–177. ACM Press, 2003.

[3] JJ Cadiz, Attila Narin, Gavin Jancke, Anoop Gupta, and Michael Boyle. Exploring PC-Telephone convergence with the enhanced telephony prototype. In *Proceedings of the 2004 conference on human factors in computing systems*, pages 215–222. ACM Press, 2004.

[4] Cisco Systems, Inc. Cisco 7900 Series IP Phones. Online at `http://www.cisco.com/en/US/products/hw/phones/ps379/`.

[5] Michael Foley. Bluetooth Special Interest Group. Online at `http://www.bluetooth.com/about/`.

[6] David Garlan, Dan Siewiorek, Asim Smailagic, and Peter Steenkiste. Project Aura: Toward distraction-free pervasive computing. *IEEE Pervasive Computing*, 1(2):22–31, 2002.

[7] N. Golmie, R. E. Van Dyck, A. Soltanian, A. Tonnerre, and O. Rebala. Interference evaluation of Bluetooth and IEEE 802.11b systems. *Wireless Networks*, 9(3):201–211, 2003.

[8] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. In *5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 1999.

[9] Trolltech Inc. Qtopia PDA edition. Online at `http://www.trolltech.com`.

[10] Terry Keeley. Thin, high performance computing over the internet. In *Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, page 407. IEEE Computer Society, 2000.

[11] Sai-Lai Lo, Duncan Grisby, David Riddoch, James Weatherall, David Scott, Tristan Richardson, Eoin Carroll, David Evers, and Christof Meerwald. omniORB. Available online at `http://omniorb.sf.net`.

[12] Lucent Technologies. Accelerate Voice over IP Solutions. Online at `http://www.lucent.com/solutions/accel.html`.

[13] Jennifer Mankoff, Anind K. Dey, Gary Hsieh, Julie Kientz, Scott Lederer, and Morgan Ames. Heuristic evaluation of ambient displays. In *Proceedings of the conference on Human factors in computing systems*, pages 169–176. ACM Press, 2003.

[14] Kieran Mansley, Alastair R. Beresford, and David Scott. The carrot approach: Encouraging use of location systems. In *Proceedings of the Sixth International Conference in Ubiquitous Computing*. Springer-Verlag, September 2004.

[15] Kieran Mansley, David Scott, Alastair Tse, and Anil Madhavapeddy. Feedback, location, accuracy: Exploring tradeoffs in location-aware gaming. In *ACM SIGCOMM 2004 Workshop on NetGames*, August 2004.

[16] Elizabeth D. Mynatt, Maribeth Back, Roy Want, Michael Baer, and Jason B. Ellis. Designing audio aura. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 566–573. ACM Press/Addison-Wesley Publishing Co., 1998.

[17] Les Nelson, Sara Bly, and Tomas Sokoler. Quiet calls: talking silently on mobile phones. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 174–181. ACM Press, 2001.

[18] William M. Newman, Margery A. Eldridge, and Michael G. Lamming. PEPYS: Generating Autobiographies by Automatic Tracking. In *Proceedings of the Second European Conference on Computer-Supported Cooperative Work*, pages 175–188, 1991.

[19] Nortel Networks Ltd. Multimedia communications. Online at `http://www.nortelnetworks.com/products/mm-communications/`.

[20] Adrian Nye. *X Protocol reference manual (3rd ed.)*. O'Reilly & Associates, Inc., 1992.

[21] Tristan Richardson, Quentin Stafford-Fraser, Kenneth R. Wood, and Andy Hopper. Virtual Network Computing. *IEEE Internet Computing*, 2(1):33–38, January 1998.

[22] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. RFC3261 — SIP: Session Initiation Protocol, June 2002.

[23] Brian K. Schmidt, Monica S. Lam, and J. Duane Northcutt. The interactive performance of slim: a stateless, thin-client architecture. In *Proceedings of the seventeenth ACM symposium on Operating systems principles*, pages 32–47. ACM Press, 1999.

[24] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RFC1889 — RTP: A transport protocol for real-time applications, January 1996.

[25] Daniel Siewiorek, Asim Smailagic, Junichi Furukawa, Andreas Krause, Neema Moraveji, Kathryn Reiger, Jeremy Shaffer, and Fei Lung Wong. Sensay: A context-aware mobile phone. In *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, page 248. IEEE Computer Society, 2003.

[26] Ripduman Sohan, Andy Harter, and Andrew Hopper. TiCL: A service architecture for thin-client networks. Technical report, University of Cambridge Engineering Department, September 2004. CUED/F-INFENG/TR.493.

[27] Jeremy Sugerman, Ganesh Venkitachalam, and Beng-Hong Lim. Virtualizing I/O devices on VMware workstation's hosted virtual machine monitor. In *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*, pages 1–14. USENIX Association, 2001.

[28] Douglas B. Terry and Daniel C. Swinehart. Managing stored voice in the Etherphone system. *ACM Trans. Comput. Syst.*, 6(1):3–27, 1988.

[29] Stephen van den Berg and Philip Guenther. `procmail` email processing tool. Available online at `http://www.procmail.org`.

[30] Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evandro Gouvea, Peter Wolf, and Joe Woelfel. Sphinx-4: A flexible open-source framework for speech recognition. Available online at `http://cmusphinx.sf.net`.

[31] Michael S. Warren, Eric H. Weigle, and Wu-Chun Feng. High-density computing: a 240-processor beowulf in one cubic meter. In *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, pages 1–11. IEEE Computer Society Press, 2002.

[32] S. Jae Yang, Jason Nieh, Matt Selsky, and Nikhil Tiwari. The performance of remote display mechanisms for thin-client computing. In *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*, pages 131–146. USENIX Association, 2002.

[33] P.T. Zellweger, D.B. Terry, and D.C. Swinehart. An overview of the Etherphone system and its applications. In *In Proceedings of Second IEEE Conference Computer Workstations*, pages 160–168. IEEE, 1988.