Ubiquitous Networks and their Applications

James Weatherall and Alan Jones

March 14, 2002

Abstract

Recent improvements in microprocessor and radio technologies have led to an increasing interest in wireless networking. At the same time, personal digital electronics have become widespread. To the enthusiast these two trends immediately bring to mind a wide range of esoteric applications. Commercially, by contrast, few substantial applications for embedded wireless networks have been proposed. This paper describes PEN, a short-range, low power wireless network specifically designed to be a ubiquitous first-point-of-contact between any and every kind of device. We describe some of the key technical issues encountered in PEN's development and illustrate these by example with applications we have built.

1 Introduction

1.1 Wireless or wire-less?

Advocates of wireless networking face a number of obstacles, not least that of providing a convincing argument that wireless networks provide something their wired counterparts cannot. The uses of networks in general, and in particular of wired networks, are many and varied and provide sufficient motivation for their continued existence. The success of the Internet has shown the potential of ubiquitous networking, accessible from anywhere at any time.

Because of their relative infancy, wireless networks lack the validation provided by such a large pool of dependent applications. Wireless networks have primarily become popular in user-interface devices, where tether-less operation provides greater convenience to the user, providing equivalent features to their wired cousins. The success of mobile telephony and wireless Local Area Networks (LANs) illustrates this.

Although tether-less access to network resources is a natural application with which to wean consumers onto wireless products, it does little to exercise their full capabilities. This paper describes several simple applications which we think show that "wireless" means more than just wire-less.

1.2 Mobile Internet Access

Mobile telephony services are perhaps the most obvious examples of widespread wireless networks. Over the past ten years, mobile phones have moved from the position of luxury item to become an everyday commodity.

At the same time, the Internet has seen a meteoric rise to fame as a consequence of increasing interest in the World Wide Web. A large proportion of homes with computers now combine these with modems to provide Internet access.

Commercial efforts such as i-Mode[22][23] and standards such as the Wireless Application Protocol (WAP)[37] have attempted to build upon the success of these two distinct technologies, by providing Internet-style services through mobile phones. In practice the services provided are often little more than email access and limited World Wide Web browsing, although this situation is improving. When combined with a lack of web content suitably formatted for display on mobile phones and archaic, telephony-oriented pricing schemes, this has not led to the rush of consumer interest that might have been.

1.3 Peer-to-peer Networking

The World Wide Web is probably the largest single Internet application, and as such is repeatedly touted as the Internet's "Killer Application". Although the Web is undoubtedly a significant motivation for the Internet's continued existence, it is by no means the only worthwhile application of the Internet.

The Web's success stems from a the fact that the Internet is a peer-to-peer network. Any host attached to the Internet can communicate transparently with any other. This means that just as any Internet-connected PC can act as a client and browse web services, it can equally well act as a server and provide its own.

The success of the Internet itself derives both from its support for peer-to-peer applications, and, more importantly, from its use of standard protocols independent of underlying network technologies. Heterogeneous Local Area Networks (LANs) may be linked seamlessly using the Internet Protocol, to allow wide-area peer-to-peer communication. At the same time, peers communicating within a single LAN are not vulnerable to failures in the rest of the Internet.

By contrast, the model of wireless networking advocated by mobile phone manufacturers is, unsurprisingly, a client-server model in which the phone is the client of services running on a backbone network. This model suits applications such as Web browsing, which can be augmented in some systems with location information to provide context. Applications involving peer-to-peer communication between devices are less well catered for, making such systems unsuitable. In particular, the requirement to communicate via a central service presents problems for "low-end" applications, such as remote control or environmental sensing.

It seems that rather than a "Killer Application", we should be aiming to provide a "Killer Feature" - the ability to communicate information between two devices, without requiring supporting infrastructure.

1.4 Short Range Wireless Networks

The ability to communicate information freely between co-located peers without physically connecting them leads to new application areas which would previously have been either impossible to implement or tiresome to use.

Several applications developed as part of the Prototype Embedded Network (PEN) project are described in the following sections. These applications could be supported by wired alternatives or other wireless networks, at the cost of convenience to the user. The applications most uniquely suited to low power wireless networks are those involving environmental sensing, whether it be used to automate other systems or simply to obtain information on environmental conditions. Sensors with wireless networking can be placed more freely than wired ones, often allowing more meaningful readings to be obtained. Since they require little power, they need not be recharged or have their batteries replaced frequently.

By combining this sensor information with the increasing number of microprocessor controlled appliances found around the home and even carried by users, we can perform onerous maintenance tasks without requiring human intervention. In order to do so, both sensors and appliances must be networked. One approach might be to connect sensors and appliances to the Internet via wired or wireless networks as appropriate. Power requirements provide compelling arguments against this approach. Mobile phone technologies rely on the fact that phones can be conveniently recharged daily, whereas simple sensors placed in discreet locations around a home or office cannot be expected to be recharged this often. At a more basic level, mobile phone technology is too expensive to embed into commodity sensors.

Environmental and personal sensors could reasonably be expected to power themselves from ambient sources such as sunlight[1], heat or movement. The power levels available from such energy sources are either consistent but minimal or are extremely variable with time. Power constraints pose a significant challenge to most wireless networks currently available.

1.5 Related Work

1.5.1 Home Area Networks

X10[13] is standard for programmable control of mains-powered home appliances through existing mains wiring.

Control signals are transmitted between devices over the electrical wiring of the user's home, using a modulated carrier signal overlaid on the normal AC power supply. This makes X10 particularly suitable for a home environment.

The control commands available are limited and there is no intrinsic verification of compatibility between a device and the control commands it is being sent. The address space for devices is limited to 256 entries which must be manually administered, which in some cases can cause clashes between devices in different homes but with the same address.

BlueTooth[3] is a recent initiative primarily supported by mobile phone manufacturers, and designed as a means to connect personal devices wirelessly. Using BlueTooth, a user's PDA, laptop, mobile phone and ear-piece can all communicate without the need for cumbersome wiring.

BlueTooth operates in the 2.4GHz Industrial, Scientific and Medical (ISM) band and provides around one Megabit per second of useful bandwidth. BlueTooth units are arranged into "piconets" of one master device and up to seven slaves. This arrangement depends upon the master device always being a relatively well-powered device such as a laptop or mobile phone.

[1] Even in the U.K.

Philips RF-Lite[31] is a home-area network with similar target applications to X10. RF-Lite provides low-bitrate bi-directional communications between up to 254 devices, of which one device must be the network master, and another the user's remote controller.

1.5.2 Ubiquitous Computing

The ParcTab experiment[36] is widely cited as the founding work in the field of Ubiquitous Computing. The project aimed to provide users with the ability to access computing resources in an untethered, mobile fashion. By using infrared as the physical communications channel, the ParcTab system also provides location information to applications, which can then be used to tailor them to the local environment.

ParcTabs are hand-held devices with three buttons, a touch-sensitive LCD display and an infra-red transceiver. A Tab will beacon at a fixed frequency when active in order that its owner can be located. These beacons and other infrared traffic are monitored by base-station transceivers in each room of a building and relayed to Agents for each Tab, running on a backbone network of PCs, which provide an interface between the Tab hardware and services and applications running on the network.

The project explored a variety of aspects of mobile, ubiquitous computing, among which were design of user interfaces for small devices, design of location-aware applications and uses of this client systems to handle mobile users. The requirement for backbone infrastructure was found to limit the systems usefulness in practice.

1.5.3 Multihop Wireless Routing

A number of recent projects have considered routing techniques for mobile wireless systems. The "Mobile Ad hoc NETwork" (MANET)[26] is an Internet Engineering Task Forces project to design scalable routing schemes for a backbone network of mobile wireless routers.

The CarNet[28] project attempts to address some of the failings of existing MANET routing techniques by adopting a grid-based location scheme, in which nodes are grouped according to geographical location. For a given address, there will be one particular node in each square of the grid responsible for routing packets for that destination device out of the square. Within each square, existing, less-scalable, routing techniques are used.

In the low power domain, the Hyphos[29] project considers routing across wireless networks consisting of a large number of short range devices, communicating as peers. A defining feature of this work is the assumption that devices will be continuously listening for data. By contrast, work on multihop routing for PEN[11] assumes that devices will spend most of their time incapable of communicating.

The WINS project[40] takes the alternative approach to providing general-purpose routing, by having nodes route at the application layer, merging results received from each other and sending them on as required. For sensor applications, the extra effort in designing application-level routing schemes can be outweighed by the reductions in traffic and hence power consumption of devices.

1.5.4 Locating Mobile Objects

For Internet Protocol (IP)[8]-based networks with minimal mobility, the Distributed Naming Service (DNS)[27] provides a basic name to address mapping, introducing a layer of location-independence. Highly mobile devices typically require mobility support at the IP layer, motivating the design of the Mobile-IP[7] protocol. Mobile-IP assigns a home-location to each mobile entity. Packets destined for mobile devices are passed to the home location, which forwards them to the device.

The Architecture for Location Independent CORBA Environments (ALICE)[5] provides location independence for CORBA applications running on mobile devices. The system assumes a wired backbone network with gateways to short range wireless networks at its periphery. Object references passed to clients are "swizzled" to change the (host, key) pairs locating the object to refer to the gateway nearest to the host device. When a device moves, the old gateway retains a forwarding gateway to the new location, and returns forwarding exceptions to clients access the device's objects. Frequently moving objects pose a problem for ALICE, since CORBA object references have limitless lifetimes and so forwarding references cannot be safely collected.

The Globe Location Service[15], developed to support the Globe object model, is built around a tree-structured network of node points with which objects register as they migrate. When locating objects, clients supply a flat, globally-unique name to the location service. This avoiding the performance bottleneck of communicating via a home location, and of maintaining immortal forwarding pointers. The primary downside is the need for a first-hop location to be stored at the base of the tree for every object.

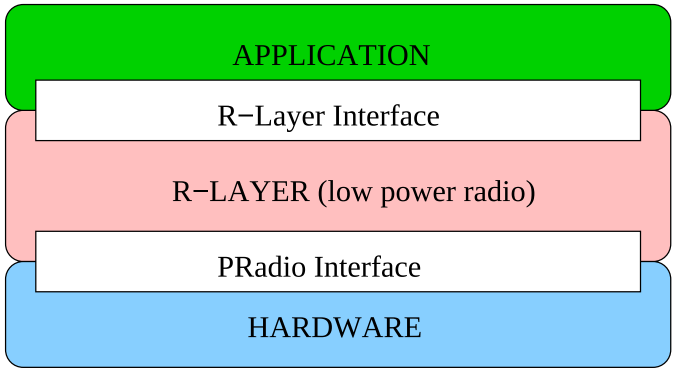
1.6 The Prototype Embedded Network

The Prototype Embedded Network (PEN)[2] aimed to research the properties of a lowest-common-denominator wireless network. Such a network would be the lowest level of discovery, rendezvous and communications for all electronic devices. It would be small enough to be embedded as a tag on books, people, tools, key-rings, wallets, etc. The more recent PicoRadio project[30] shares similar target applications.

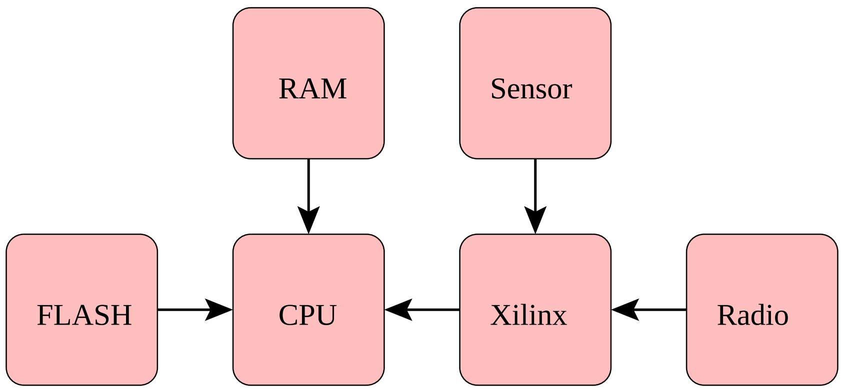
PEN cannot rely on the presence of other services for its operation. Special-purpose and high-bandwidth networks such as BlueTooth[3], WaveLAN[38], and other IEEE 802.11[20][21] networks would be activated only after first negotiating their use via PEN, as would computational, sensing and actuator resources. In many low power devices, PEN will be the only service to remain powered when the device is otherwise "off".

To achieve this, PEN nodes must be small, low-cost and low power. Nodes must be small enough to be embedded in even very tiny devices without altering their design, and cheap enough not to affect their cost significantly. Most importantly, PEN nodes must operate for prolonged periods on limited power. This motivates the use of short range radio system, capable of transmitting and receiving data with minimal power consumption. In addition to low power hardware, PEN employs specially designed low power protocols to further reduce power consumption.

The PEN devices developed at AT&T Laboratories Cambridge comprise



a) Software structure of a simple PEN device



b) Hardware structure of a simple PEN device

Figure 1: Structure of a typical PEN sensor

cheap off-the-shelf processor, radio and memory components, plus extra hardware to aid development. The prototypes operate in the 418MHz ISM band and provide 24Kbps peak bit-rate. The prototypes are larger and consume more power than we would like, but provide an excellent platform for application prototyping. A commercial single-chip version of the system would allow substantial reductions in size, power consumption and per-unit costs.

Figure 1.6 shows the structure of a typical PEN sensor. In addition to the components illustrated, all PEN devices will contain some non-volatile storage for a fixed unique identifier and human-readable description of the device.

Figure 2 summarises the characteristics of our prototype units[25] and those we expect from production equivalents.

The defining feature of PEN is its low power data transfer protocols. These operate by causing devices to spend most of their lifetimes in dormant "sleep" states, unable to communicate, waking only infrequently to rendezvous with

Feature	Prototype	Production Device	Production Sensor
RAM	128K	32K	<1K
FLASH/ROM	256K	32K	<4K
Size	10x5x2cm	2x2x0.4cm	1x1x0.2cm
Power (peak/avg.)	300mW/1mW	4mW/6μW	4mW/6μW

Figure 2: Characteristics of PEN prototypes and estimates for production units

others. This assumption is valid for a wide range of applications and is used to great effect by pager systems such as POCSAG[4] and FLEX[10]. These systems achieve battery lifetimes of up to a year by synchronising low power clients with high-power transmitters at regular intervals.

Some short range ad hoc network technologies, such as RF-Lite and BlueTooth, take a similar master-slave approach, requiring all communication between slaves to be scheduled by the master. The master can schedule communication to make best use of the channel and to provide quality of service guarantees but must expend more power than its slaves in doing so.

PEN's protocols are designed for decentralised communication, allowing peers to communicate without infrastructural support. Devices rendezvous in sender/receiver pairs and agree suitable transmission patterns according to their own particular requirements and observations of local radio traffic. This can be used to make greater power savings than with a centralised master-slave design.

The PEN hardware and core system services have matured to become a stable platform on which to prototype applications. Most current development for PEN is therefore in applications research, particularly by members of the affiliated Laboratory for Communications Engineering, Cambridge University.

2 Home Automation

The home of an enthusiastic AT&T Labs employee has been fitted with Programmable Interface Controller (PIC)-controlled lights and curtains. Each PIC controls an RS232[9] serial interface through which sequences of commands may be sent and responses received. Each PIC may also be programmed with a schedule of commands to execute at specific times, to allow stand-alone operation.

The home is also equipped with a hi-fi and television set. Infra-red transceivers in most rooms allow conventional remote controls to be used, and infra-red devices to be controlled.

2.1 The Peripheral Approach

The simplest way to arrange out "active home" is to treat each device as a peripheral of a central controlling computer. The computer then communicates with devices using their own proprietary protocol via bus technologies such as RS232, or USB[34], for example. All communication between devices passes via the controlling computer, as illustrated in Figure 3.

Since this approach requires all devices in the home to be wired back to a single location, adding new devices can involve excessive effort. Devices will be unable to communicate with each other if the central controller is not available

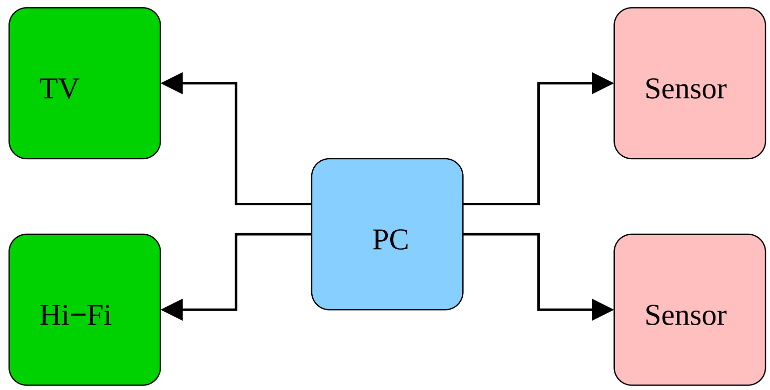


Figure 3: Home appliances connected as peripherals to a central controller

for some reason[2]. The X10 specification[13] tackles this problem by providing low bit-rate communication between devices over existing mains power lines.

2.2 The Wired Approach

Using a wired networking, we wire devices such as lights and curtain-controllers together as peers, via a shared network as shown in Figure 4. For small numbers of devices, we might use a central PC as a "hub", as for the peripheral approach. Larger homes would use a fully switched network and may even have multiple controllers to provide fault-tolerance.

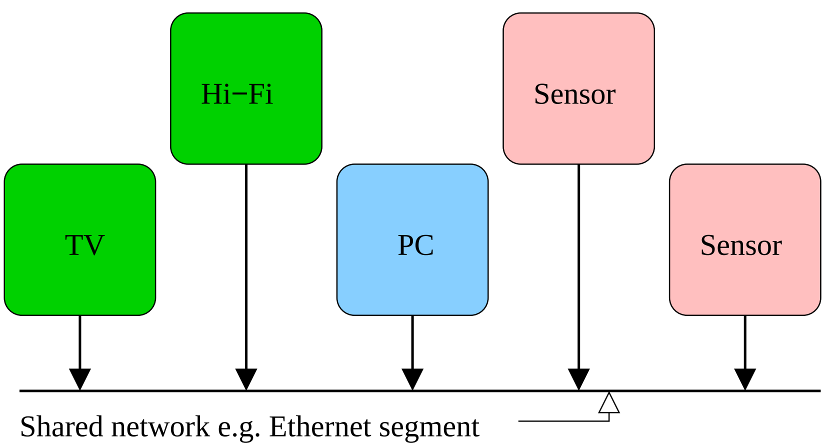


Figure 4: Home appliances connected to a peer-to-peer wired network

Use of a shared wired network is favoured by home multimedia-oriented projects such as Warren[6] and HAVi[16][17], which use ATM[1] and IEEE Std

[2]A problem alluded to in presentations on the HomeAPI system

1394-1995 “Firewire”[19] physical networks, respectively. The HAN[14] and AutoHAN projects build on the Warren network to support ubiquitous computing applications, illustrating the benefits of general-purpose networking in home environments.

Wired architectures work well for multimedia applications but are less well-suited to user-interface and sensor devices. Such devices could instead be equipped with wireless networking and attached to the wired network via intermediaries. This poses problems of compatibility between wireless devices and of reliability of the intermediary device(s).

2.3 The Wireless Approach

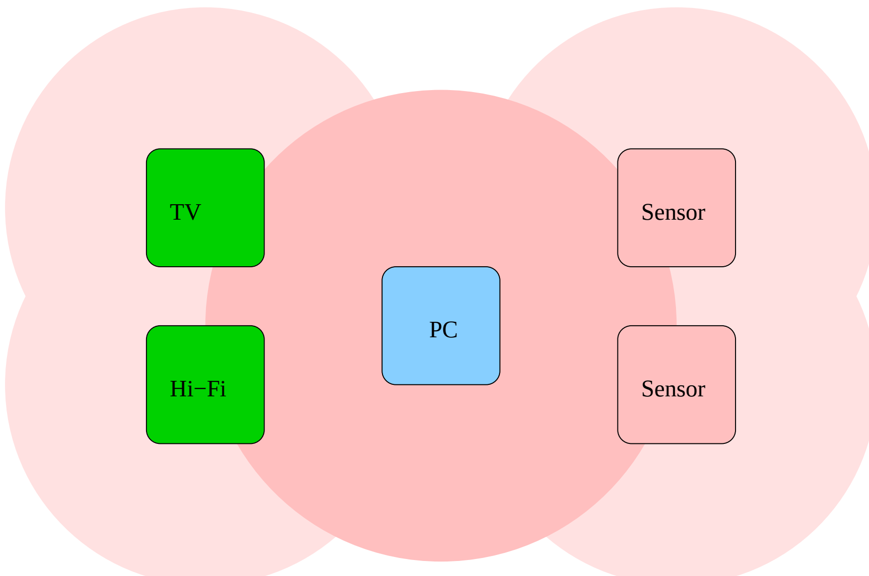


Figure 5: Wireless appliances communicating via a central master

We have taken the approach of equipping all devices with their own embedded PEN node. Our PIC lighting controller, for example, has a PEN node attached to the serial port which understands the controller’s serial protocol and translates between this and a more standard radio protocol. Light and temperature sensors are similarly equipped, again using standard protocols over the radio.

A wireless system of this kind can be arranged in one of two ways, as illustrated in Figures 5 and 6. Philip’s RF-Lite network, for example, takes the centralised master-slave approach, with one device acting as a hub for all wireless communications. By contrast, in using PEN we take a completely peer-to-peer approach. This gives us greater freedom in deploying devices, but presents further technical challenges which we will discuss in section 4.

Multimedia appliances in our home are connected with a single 10Mbs Ethernet-compatible network. Initially, PEN seems redundant with such well-connected devices. We envisage PEN as a transport for small amounts of sensor

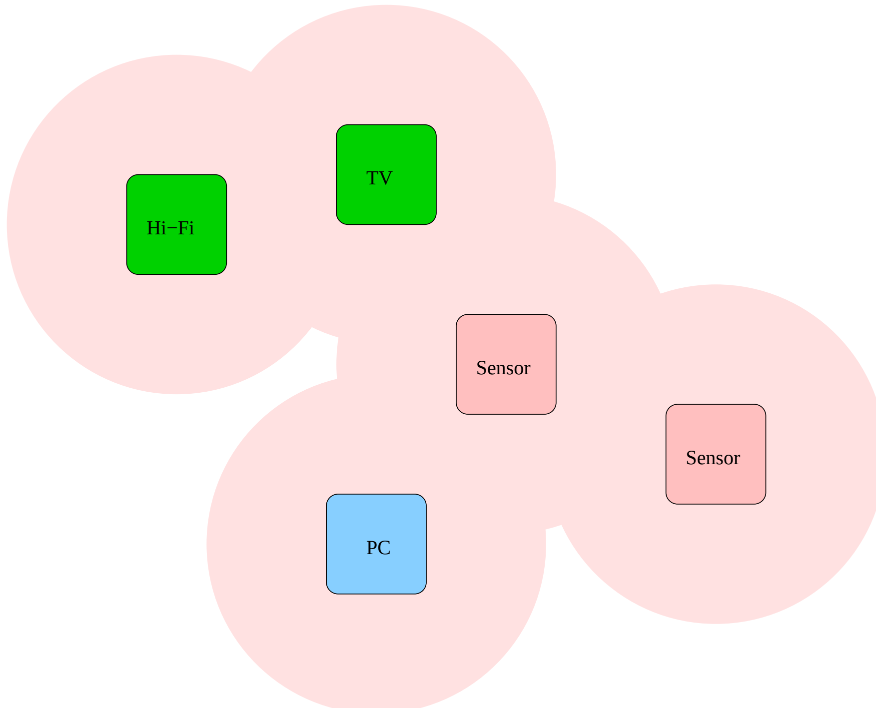


Figure 6: Wireless appliances communicating peer-to-peer

and control information, however, rather than as a general-purpose network. PEN is therefore used to configure and monitor high-power, well-connected equipment even though the actual transfer of data might take place over Ethernet or WaveLAN.

2.3.1 Big Brother

For convenience, we we keep a central PC responsible for day-to-day maintenance and configuration of the other devices in the home, as in the peripheral and wired models. The PC processes sensor data and user input from around the home and manipulates devices according to user-defined rules.

2.3.2 Thin-Client Control

A number of thin-client wireless user interface devices are deployed around the home. The central home controller can watch for these devices and render a user interface whenever one becomes available. The intelligence behind the interface resides in the central controller, which accepts input events from the device and replies with graphics to be rendered. Through the user interface the user may either opt to take direct control over available devices, or to modify existing settings for automatic operation. Each display may also be made to default to a particular purpose, such as acting as a hi-fi remote controller or alarm clock.

Thin-client wireless user interfaces have been researched in the past, most notably by the ParcTab project[36]. One of the principle difficulties encountered was due to the high power consumption required to maintain a responsive user interface in the ParcTab system. We take advantage of common temporal characteristics of user interactions to schedule sleep intervals intelligently. When the user presses a button, for example, any well-designed user interface will change to reflect that the action has been registered. The interface device can therefore wait a short while after each input event for updates to the display, to avoid introducing excessive latency.

2.3.3 Ad Hoc Control and Configuration

In the peripheral model, all devices in the home were attached to a single central controller which would act as a hub for communications between devices. Clearly, this presents a single point of failure. Should the central controller fail, no appliance in the home will be able to communicate with any other. Devices will be forced to revert to stand-alone default behaviours.

The wired model allows us to eliminate the need for a central controller - suitably equipped devices can interact directly across the network. Alternatively, we can also attach backup controllers to the network, which can take over if the main controller fails.

Using a low-power wireless network, we can go one step further. By attaching a PEN node to a PDA, we can take direct control of devices in our local environment. In fact, a typical PDA is more than capable of controlling a home-full of devices in its own right.

Because all devices can communicate as peers, including simple sensors, appliances can be configured with simple rules and left to operate without further attention. The lighting controller can be configured to "watch" switches and sensors around the home and to operate lights accordingly. Similarly, a switch can be configured to operate a particular light directly. If the central controller fails then both systems will continue to operate independently.

Since devices can be configured to operate independently of a central controller using simple rules, a controller is only required during the initial setup of the system. We can avoid then need for a controller at all by equipping a telephone with PEN, for example. Devices may then be configured remotely by the manufacturer or retailer. Similarly, a television set or other user-interface device could be equipped with enough intelligence to take direct control over devices. If a PEN-telephone is available, the television can download scripts to control other devices, direct from their manufacturers, to provide standard configurations. We avoid abuse of the ability to configure devices remotely through use of security protocols such as the Resurrecting Duckling[33].

3 Application Classes

Some devices in our home are controlled automatically, without direct action from the user, by their PC or PDA. Others must respond to ad hoc control requests from the user in a timely fashion. Both forms of interaction must be catered for efficiently by a single low-power network. To do so requires that each application be allowed to specify its own requirements for the delivery of datagrams.

Properties such as timeliness, caching, data fusion, and feedback, for example, are important distinguishing characteristics between applications. While applications involving direct user interaction will often require extremely low-latency responses, more automated applications may trade timeliness for improved efficiency. Delivery concerns are therefore left to application level protocols, which then build upon the standard PEN primitives for discovery, rendezvous and messaging[12][24].

In the home, most devices will be able to contact each other directly. If a direct connection is not possible, data may be exchanged using multihop techniques or via a static backbone routing mesh. Routing algorithms must consider power management issues and deal with variable link quality, caused both by changing conditions and by device mobility. Again, the tradeoffs involved are often best dealt with at the application level.

To illustrate the distinctions between different application domains with respect to delivery and routing characteristics, we present three example scenarios, called “Switch”, “Trigger”, and “Logger”.

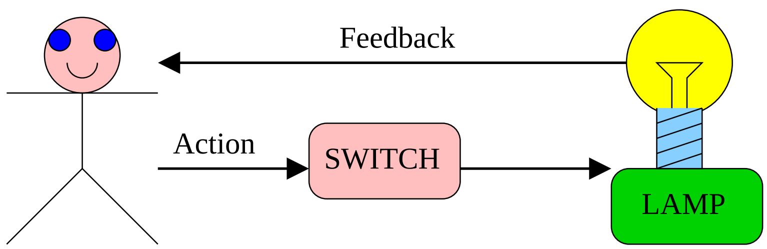


Figure 7: Using a switch to toggled a lamp and seeing the response

3.1 Switch

Switch applications are those in which a signal from a user must cause one or more devices to activate, with no perceptible latency, as illustrated in Figure 7. In practice this means less than around 100ms latency.

3.1.1 Light Switches

Light switches are an example of this. If the switch and device are not too far apart then we may communicate directly between them. The light can remain active at all times, since it is mains powered. The switch remains dormant unless mechanically activated, whereupon it transmits its new state to the light before returning to sleep.

Provided the lamp has power and the radio is not jammed, the switch can control it regardless of the state of other devices in the home. On the rare occasions in which a command from the switch fails, perhaps because the light is temporarily without mains power, it can periodically retry the operation until successful.

3.1.2 Doorbells

Doorbells will often differ from lights in that they may be out of range of the button which triggers them. We must therefore route button press messages through other PEN nodes to bell sounders. Ideally, these routing nodes would remain fully powered at all times, and would retransmit commands immediately after receiving them.

If several hops are required then we must select a route determination algorithm for the routing nodes to use. This may be as simple as flood routing, or may consider link qualities, power resources and latencies in order to conserve bandwidth and energy.

If we wish to route data through low power nodes then we instead use store-and-forward messaging. Nodes synchronise in pairs and rendezvous with each other at regular intervals to exchange routed messages. To stay synchronised, nodes must occasionally transmit a "beacon" datagram, even when they have no messages to forward. Low power routing nodes might wake to rendezvous every ten milliseconds, for example, leading to an average latency per hop of only five milliseconds. Given a direct radio range of five to ten metres, a doorbell signal can traverse even a large building in under a tenth of a second.

3.2 Trigger

Trigger applications are those in which a switch is used to send a command to one or more devices, with relatively relaxed timing constraints. A switch near the exit from a room or building, for example, can command all selected lights and devices to switch off, as shown in Figure 8. The command can take several minutes to perform without causing ill-effects, using timestamping to avoid overriding more recent commands.

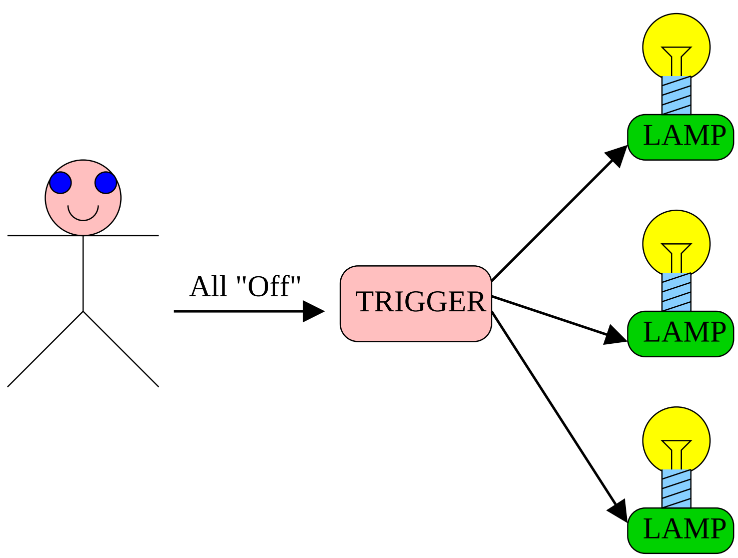


Figure 8: Triggering an action to start

In this case the routing nodes could rendezvous only every thirty seconds, for example. This is sufficiently infrequently that virtually any PEN device could therefore take part in routing.

3.3 Logger

Around our home, we have distributed a set of sensor nodes to measure the ambient temperature. Each sensor measures the temperature at regular intervals and stores a time-stamped sequence of readings.

When we require the readings, a "logger" node is carried around the building to collect them. Sensors upload their readings to the logger as it moves within range of them, as shown in Figure 9. Later, the logger is placed near a download station, which transfers the new readings to safe storage and deletes them from the logger.

The download station might be connected directly to a PC, or feed readings back to a central storage location via a network. The logger functionality could be provided in a standard PDA, or as part of any other electronic device the user carries.

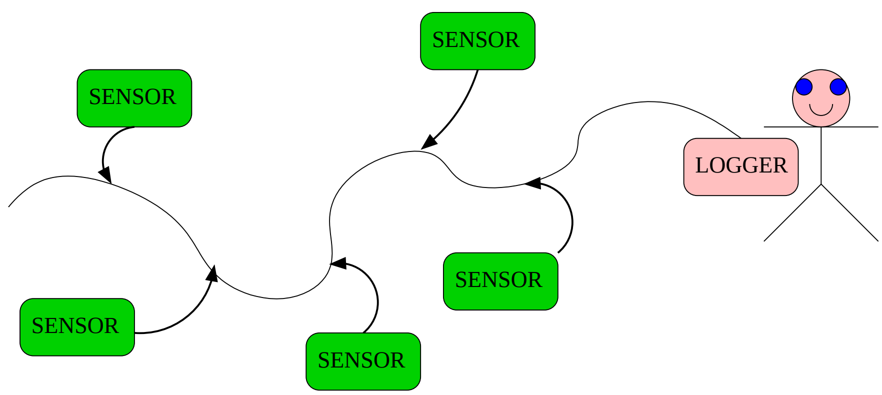


Figure 9: Logging sensor readings

The next time the logger encounters these sensors, it first downloads the latest set of readings. It then instructs each sensor to erase the old readings, since they are now safely stored by the download station.

This kind of logging is ideally suited to industrial contexts where sensors are distributed over a wide area, relative to the radio's range. Even in a home environment we might wish to collect readings from a nursery, fridge, greenhouse, pond or garden, and from appliances around the home. In the home, there is unlikely to be a need for frequent synchronisation of sensor data. A wearable device such as a mobile phone could therefore be used to download readings whenever they are available, as the user moves around their home.

Sensor nodes transmit short beacons datagrams at regular intervals (every 30 seconds in our experiment), and listen for a short period after each interval for incoming data. Nearby loggers listen for beacons in order to discovery available sensors. When a sensor is discovered, the logger will inform it of which readings it has already logged and which have been stored, allowing to the sensor to avoid re-transmitting logged readings, and to delete stored ones.

To save power, the logger can be explicitly triggered to listen for beacons for a few minutes by a button press. An alternative is to assume that loggers will be recharged frequently and that they can therefore listen continuously for sensors. The PEN logger prototype takes the former approach, requiring the user to trigger sensor-gathering operations. Because this requires user intervention, the user may be informed with a beep when all available data has been transferred and it is safe to move on to other sensors.

This is an example where there is no explicit routing, because the data itself is being transferred from one high level object to another as it travels, rather than packets being routed unchanged from one node to another. Another example of this is where data is combined in local nodes before onward transmission, a technique that can conserve both bandwidth and power as in the WINS project[40].

4 Routing

In a typical home or office environment devices may be distributed over a wide area. Centralised wireless systems typically rely on the presence of a powered master device capable of transmitting and receiving data over a wide enough range to reach every slave. Since we wish to support ad hoc communication using peer-to-peer networking, we cannot assume that all devices will be accessible in a single hop.

This makes it impossible to issue blanket configuration requests, such as a "trigger" to power down all active lighting or heating in the home, without performing some sort of routing of packets. There are many groups researching wireless routing protocols, in particular the MANET group of the IETF. While such efforts are mostly aimed at supporting wireless IP systems, the techniques described below are designed with low power, short range, embedded radio systems in mind.

4.1 Wormhole Routing

We could simply attach several radio nodes to a wired network, and arrange for all packets seen by one node to be retransmitted by the other nodes. All packets, be they discovery beacons, rendezvous packets or datagrams, would be retransmitted. In effect, the radio spaces around the linked nodes would be combined into one. This arrangement, shown in Figure 10, is dubbed a "wormhole".

4.2 Tagged Wormhole Routing

The next level of sophistication would label packets to indicate whether they were permitted to traverse wormholes, or whether they already had. The latter flag is a minimum requirement in practice, since it allows infinite packet retransmissions to be suppressed. Infinite retransmissions might occur if two wormholes were set up between the same pair of radio spaces. It also warns

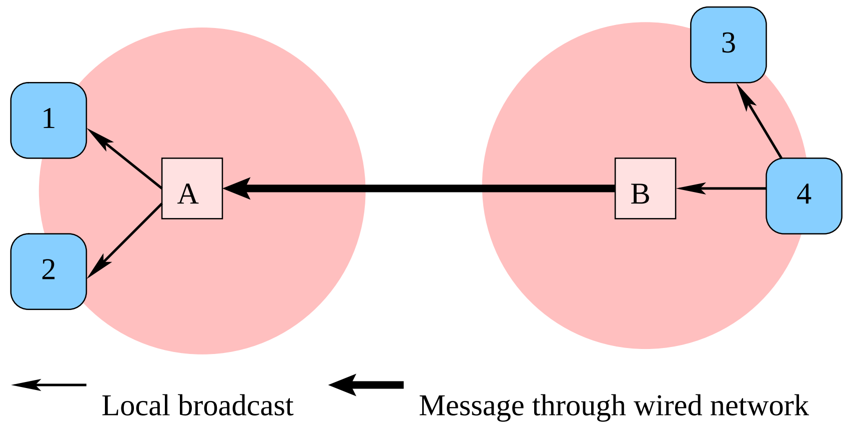


Figure 10: Wormhole routing of wireless packets through a wired network

nodes that the usual relationship between radio connectivity and proximity has been broken.

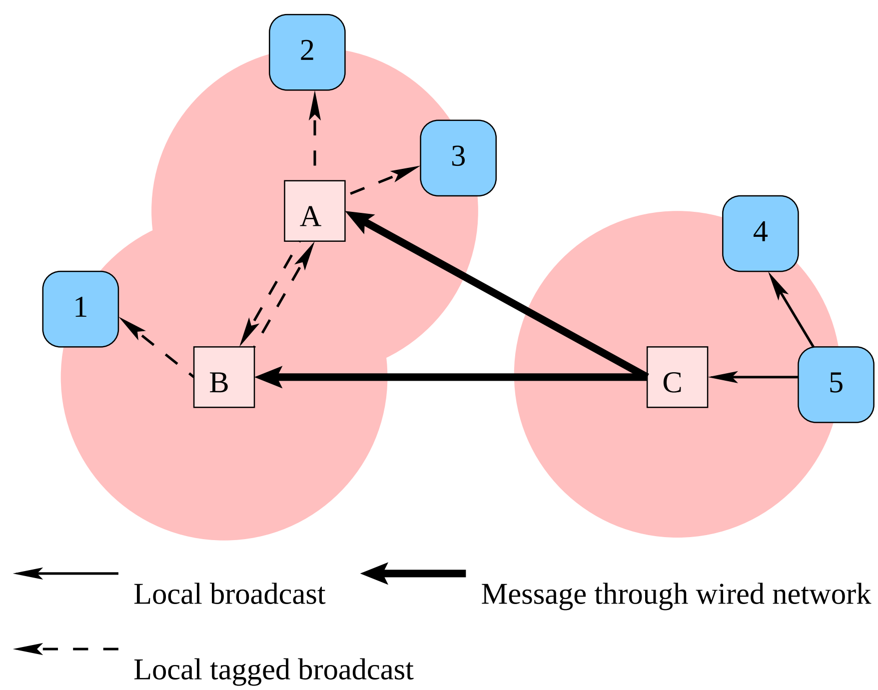


Figure 11: Tagged wormhole routing of wireless packets through a wired network

Figure 11 illustrates tagged wormhole routing. The message transmitted by node 5 is retransmitted remotely by A and B, whose radio domains overlap. This would present a problem for basic wormhole routing, since the retransmitted packets would then be passed back across the network by the B and A.

4.3 Other schemes

Wormhole routing allows short range wireless devices to communicate over arbitrary distances via wired infrastructure. They do not address the issue of high mobility of wireless devices, or of scalability to large numbers of wormhole gateways. We can forge a wormhole route between a user's office and various rooms in their home, for instance, but we then suffer problems when that user roams around the building with their wireless devices.

Other modifications to improve the performance of wormholes include use of multicast nets. We might connect all gateways in an office building to the homes of all employees. Each user would then be assigned their own multicast group, so that only packets from their devices would be transmitted to their home.

A minor variant of this technique would be to allow the user's devices to specify specific the gateways through which their packets should be retransmitted, by wired network address.

4.4 Wireless-through-Wired Switching

The techniques described above are ideal for certain applications, such as feeding sensor readings from a particular location to a sensor logger via a wired network. They cannot, however, support routing of data between arbitrary devices on demand over the wide area. We have therefore designed and built a switched routing service known as R-Link. R-Link allows unicast packets from a source device to be delivered transparently via a wired network to their destination.

The service resides in powered PEN gateways attached to a backbone wired network. When a PEN device fails to transmit a message to the destination device directly, it falls back to requesting the well-known "R-Link" service locally. Any nearby R-Link gateway will respond to the request and route the datagram through the wired network to the target device.

To route a datagram from one device to another, the R-Link service must be able to locate a gateway within range of the target device. This is achieved using Predator, a scalable distributed location service[39], which runs in the wired network. R-Link gateways continuously listen for beacons from nearby PEN devices. When a beacon is heard, the gateway registers itself with the location service as a contact point for the device. Subsequent attempts to transmit data to the device result in a location service lookup for a suitable contact gateway. Packets are then forwarded directly to the gateway, to be forwarded to the target device.

Figure 12 shows a typical search for an R-Link gateway through Predator. Node A ("/AT&T/Node20") attempts transmit data via a gateway to B ("/LCE/Node1"). Node names are hierarchically assigned, so the search is initially for the "LCE" subtree. When an "LCE" subtree node is found, it is then asked to search within the tree for "Node1". Note that the hierarchical arrangement means that the root node "R" need not be aware of the names or locations of "Node1" or "Node20", only of the subtrees they belong to, reducing the load on it. In practice, "Node20" would start the search from an "AT&T" subtree node, which we have omitted for clarity.

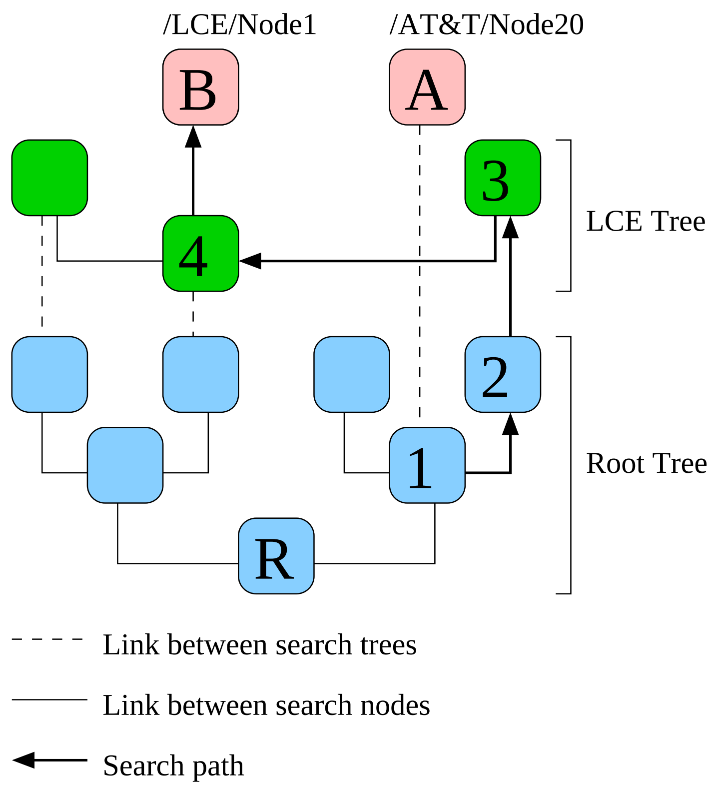


Figure 12: An example search using Predator

Low power devices may use powered routing gateways as "virtual batteries", by requesting that they route packets even when the target device is directly contactable. Direct transmission would require the sender to listen for a beacon from the receiver before sending. The client can instead pass data to the gateway and immediately return to sleep, since transmission to powered devices is immediate. In spite of the extra hop, most messages will be delivered no later than they would have been if sent directly.

4.5 Wireless-to-wired Routing

The R-Link service described above deals exclusively with targeted datagrams. However, the PEN primitives also allow for datagrams to be sent to any local instance of a particular service. These datagrams are more difficult to route through a wired network, since applications often use them to infer context information, relying on the fact that they will only be delivered to local nodes.

There are, however, situations in which it is desirable to host services in the wired network. This requires that they are advertised by gateways to nearby nodes. PEN devices include an APplication IDentifier (APID) in their beacon packets for each service they provide. A wide-area network may contain hundreds of services, making this form of advertisement infeasible.

Equally, there may be hundreds or thousands of gateways into the wired network. It is infeasible to expect services to contact every gateway to request that they be advertised.

The APIDgate service advertises services on demand, in response to "plea" packets from PEN devices. Plea packets are used to reduce latency when communicating with powered devices. When a gateway receives a plea packet, it uses the Predator location service to lookup the desired APID. If a service using the APID is found then the APID is added to those advertised by the gateway. This addition implicitly causes the low power radio layer to broadcast an "offer" packet for the APID. Since the client device will be awake for several seconds, listening for beacons, it will hear this offer and take advantage of it.

Figure 13 shows a PEN application using a service via APIDgate. A PEN application requests a particular service. APIDgate sees the request and uses a Predator search to satisfy it. The application then passes data via APIDgate. When the service wishes to reply, it locates the APIDgate node nearest the PEN device and forwards data through it.

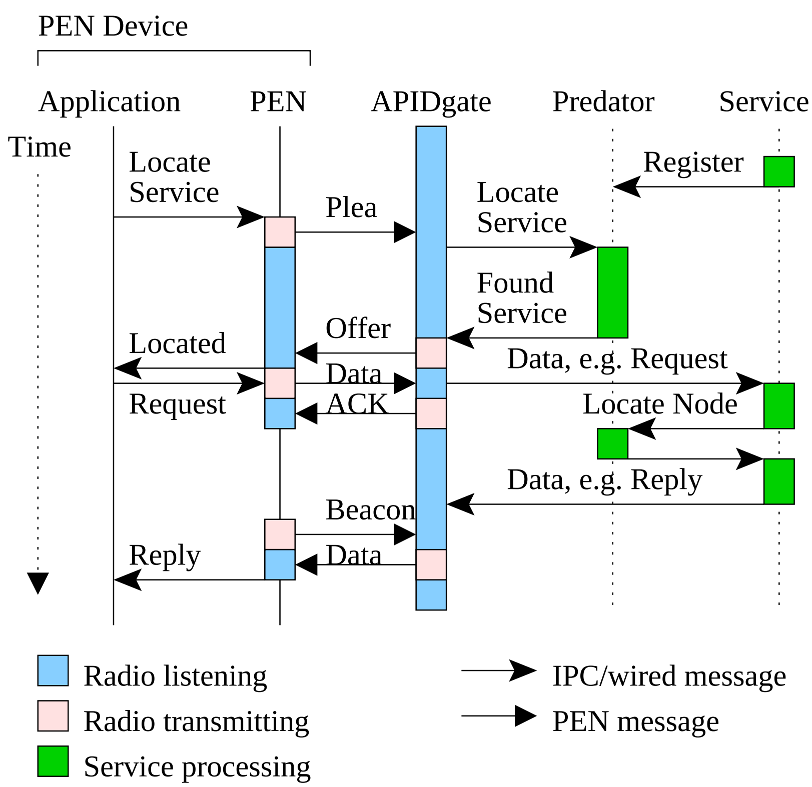


Figure 13: A PEN node using a service via APIDgate

The APIDgate service can be used to provide similar services to those available from mobile Internet technologies. Services can be provided to proxy between HTML content and a more compact form for mobile devices, for example. Services can also be provided transparently by other PEN devices, however, allowing many applications to operate both with and without supporting infrastructure.

Finally, since the R-Link service is implemented using a well-known APID, it can be provided in the wired network via APIDgate like any other service.

5 Further Experiments

5.1 Environmental Monitoring

AT&T Laboratories Cambridge runs a number of server machines in specially cooled machine-room. To avoid damage in case of failure of the cooling system, each machine monitors its own temperature and shuts down if this exceeds a specified threshold.

In mid-February one year, a time of year when most of England enjoys temperatures of below five degrees Celsius, a particular server began shutting itself down, apparently due to overheating, on a regular basis. Inspection of the machine-room's cooling system revealed no obvious cause for the failures. Temperature sensors in the cooling system itself and in other PCs around the problem machine showed consistently moderate readings.

As a last check before beginning the length task of replacing what seemed to be faulty hardware, PEN temperature sensors were placed in the machine room. Sensors were placed around the failing server and, to provide control readings, around the cooling system and other PCs in the room. Readings were collected periodically by a logger device via PEN and downloaded to a PEN-enabled PC running plotting software.

By adjusting the PEN nodes to be in specific locations, it was established that some new installed in the room equipment had altered the airflow around the problem machine and that the machine itself was not at fault. In this case, PEN helped identify the problem simply by virtue of nodes' form factors - conventional temperature recording equipment was large enough to affect the air flow and thus make it impossible to track the problem.

5.2 Active Doorsigns

As we have previously mentioned, a number of PEN nodes have been constructed with an attached display and a number of buttons. Several of these displays have been used as active doorsigns for offices around our building. Each doorsign runs a display manager and a number of small application processes, providing a number of features.

The applications described here use APIDgate and R-Link to access infrastructural services, allowing doorsigns to be easily moved and re-assigned to new tasks.

5.2.1 Location

A common problem, particularly in technology-oriented offices, is that many personnel spend a great deal of their time in other offices, or in hardware labs, for example. The Active Badge[35] system goes some way towards solving the problem by tracking the location of infra-red badges worn by personnel and equipment. Visitors can check the badge system from their PC before setting off to ensure the person they are visiting is available, but by the time they arrive that person may have gone somewhere else. Visitors unfamiliar with the badge system may not think to check in the first place.

The default application displayed on doorsigns is therefore the names and locations of all personnel normally found in the associated office. Even extremely technophobic visitors can go to a person's room and, if they are not there, look at the door sign to see where they are.

5.2.2 Appointment Management

Being a research laboratory, we often have visitors arriving to see project demonstrations. Visitors are usually given an Active Badge at reception to allow them to be contacted if necessary during their stay. In addition to their badge, each visitor can also be allocated a PEN node with their itinerary stored in it. Ideally, this would be part of their badge.

As visitors move around the building, they will be able to obtain information on their itinerary by standing in front of a doorsign display and pressing a button. The doorsign then listens for nearby appointment stores and displays a list of those it finds, allowing the user to select theirs. Given the display they are using and the appointment information in the store, they can even be told which direction to go in to get to the right place.

We could have prototyped this application by tracking visitors' Active Badges and correlating this with the locations of displays to establish which appointments to display, from a central server. The advantage of the personal-store approach is that the appointment store can reside in any PEN device the user would normally carry, such as a mobile phone or PDA, avoiding burdening them with extra equipment. Further uses of this technology are discussed in section 6.2.

5.3 Context-Aware Personal Information Management

By way of demonstration, a context-aware To-Do list application has been built for the Compaq iPAQ 3600 "PocketPC" platform. The device uses wireless LAN connectivity as a location metric, enabling it to distinguish between the Laboratory for Communications Engineering and the AT&T Laboratories Cambridge, for example. While many office-oriented context-aware applications[32] require fine-grained location information to be effective, our crude location metric works surprisingly well for this application.

We could enhance the accuracy of location information trivially by detecting the availability of Active Badge services and using those to obtain context information where available. Both wireless LAN and Active Badge location information requires the presence of installed server infrastructure, whereas the addition of PEN, besides drastically reducing power consumption, could afford location information based on the presence of well-known peer devices. As with wireless LAN schemes, we could take advantage of badge services where available.

6 Target Applications

6.1 Safety Monitoring

As well as light, humidity, temperature and other environmental sensors primarily used to maintain a comfortable living or working space, similar sensors are used to perform safety critical monitoring.

Carbon Monoxide and smoke sensors equipped with PEN nodes can signal to sounders in each room of a house in addition to their normal, in-built alarm. When their batteries become low, they will still have enough power to send periodic "battery low" indications to the nearest powered device they can find. Battery low messages will propagate between powered devices and be passed to any user interface devices through which the user may be informed, such as a TV or PDA.

6.2 Personal Information Stores

By carrying a personal PEN device, a user can keep frequently used information with them at all times, in a single, unobtrusive device. The information store might hold appointment information as previously discussed, as well as address books, notes and messages.

The user's mobile phone would backup text messages to the personal store, allowing them to be more conveniently read on any available display. When making calls on public phones the store would be again be referred to. Business cards could be broadcast to nearby recipients by simply pressing a button, or, for the paranoid, transferred via physical contact.

All information in the store could be kept in sync with one or more desktop systems, allowing it to be securely backup up or used in desktop applications, wirelessly, without user intervention.

6.3 Context Aware Configuration

A user can benefit from context aware features of devices around them. When a powered device receives a beacon message from a user's personal node, it can send a short reply message. This allows the personal node to keep track of its local context, in order to perform context-specific automatic actions the user has specified.

Management of context-aware actions could be performed via a To-Do list interface similar to that described in section 5.3. Scripted actions as well as simple text reminders may be specified, which will be ordered according to context. While most actions would require explicit initiation by the user, some, such as backing up data or downloading mail, would be automatic. The Lcron[18] project uses a similar approach, by augmenting the standard "cron" tool to use location as well as timing information.

The same personal store used to provide context-specific configuration options could also provide context-aware PIM services and act as their personal information store. Devices ranging from PDAs down to simple rings or buttons could perform the store function, allowing the user to choose the most appropriate device for them. Multiple stores belonging to a single user can synchronise transparently whenever co-located to allow them to be used interchangeably.

6.4 Power Saving

A typical home contains a wide range of electronic appliances. These range from entertainment systems such as televisions and hi-fis, to appliances such as washing machines and cookers. Similarly, in office environments there are usually many PCs, air conditioning units and lights. Most of these devices can safely be powered down when there is no-one present.

Home appliances are often left on simply because they are forgotten or because it is too much effort to go around checking that they are all off. Office appliances are treated similarly but in the case of PCs, are often left on so they may be accessed from home.

With low power wireless networking embedded in these appliances, much of the inconvenience can be taken out of simple power saving tasks. Using short range radio, devices can automatically enter low power states when a user is not present. A house-controller can be used in the home to tell all non-essential devices to power off when the owner goes to bed. At this point, the wireless portion of the system can remain actively monitoring movement sensors in the home, in order to operate lighting the moment the user requires it.

7 Conclusion

Wireless networks are becoming increasingly popular with consumers, primarily in the use of mobile phones. The success of mobiles phones stems from the freedom for two people to communicate without being tethered to a particular location. Their popularity has led to much commercial interest in supplying services to further "hook" users on wireless systems.

We have taken the view that it is the basic ability to communicate data seamlessly between two points that has led to the success of both mobile phones and the Internet, and that suitably designed wireless networks can provide features unavailable from wired systems, which lead to new types of application.

Using the home of an enthusiastic employee as a test environment, we have described some of the applications we envisage for short range and low power wireless systems, and the problems they face. In prototyping a set of typical applications, we have shown not only the advantages of using a low power wireless network to manage control and configuration of arbitrary devices, but also the importance of flexibility in that network.

The applications we have prototyped have diverse timeliness and power requirements. This motivates the use of low power protocols which can be tailored to applications' needs dynamically, rather than being controlled by a central authority as in several similar systems.

Finally, we might conclude by noting that the tradeoffs and side-effects of low-power operation often contradict conventional thinking in systems design. The introduction of microprocessors into many areas of consumer electronics has generally proven successful because the cheapest, smallest and easiest to fabricate units are typically used. If wireless networks are to be used for more than providing mobile Internet access then a similar shift from computer- to computer-oriented design must take place.

8 Acknowledgements

The authors wish to thank the AT&T Labs Cambridge and the Engineering and Physical Sciences Research Council for funding this project, and the members of the PEN project for their fine work. Thanks are also due to Sai-Lai Lo, Andy Hopper, Andy Fisher, Paul Osborn and Gray Girling for their support and suggestions.

References

[1] The ATM Forum. At `http://www.atmforum.com/`, 2001.

[2] Frazer Bennett, David Clarke, Joseph B. Evans, Andy Hopper, Alan Jones, and David Leaske. Piconet - embedded mobile networking. *IEEE Personal Communications, Vol. 4, No. 5, pp 8-15*, October 1997.

[3] Specification of the Bluetooth System Version 1.0A. At `http://www.bluetooth.com/`, July 1999.

[4] CCIR / Post Office Code Standardisation Advisor. *CCIR Recommendation 584, Radiopaging Code 1*, February 1981.

[5] Raymond Cunningham. Architecture for Location Independent CORBA Environments. Dissertation for MSc in Computer Science, Trinity College, Dublin, September 1998.

[6] D.Greaves and R.Bradbury. Warren: a Low-Cost ATM Home Area Network. *IEEE Network Magazine*, January 1998.

[7] C. Perkins (editor). IP mobility support. RFC 2002, October 1996.

[8] F. Baker (editor). Requirements for IP Version 4 Routers. RFC 1812, June 1995.

[9] Electronic Industries Association (`http://www.eia.org/`). *EIA232E - Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*, D edition, January 1987.

[10] FLEX Technologies. At `http://www.motorola.com/MIMS/MSPG/FLEX/`.

[11] Gray Girling, Jennifer Li Kam Wa, Paul Osborn, and Radina Stefanova. The design and implementation of a low power ad hoc protocol stack. In *IEEE Wireless Communications and Networking*, sept 2000.

[12] Gray Girling, Jennifer Li Kam Wa, Paul Osborn, and Radina Stefanova. The pen low power protocol stack. In *IEEE International Conference on Communications and Networks*, oct 2000.

[13] Timothy C. Green. FAQ:comp.home.automation (Version1.09). `http://www.cs.ualberta.ca/~wade/HyperHome/Faq/Faq_outline.html`, February 1995. Also other informal sources.

[14] Home Area Networks. At http://www.cl.cam.ac.uk/Research/SRG/netos/han/, June 2001.

[15] Franz J. Hauck, Maarten van Steen, and Andrew S. Tanenbaum. A location service for worldwide distributed objects. Technical report, Dept. of Math. and Computer Science, Vrije Universiteit, Amsterdam, The Netherlands, 1997.

[16] HAVi, Inc. *The HAVi Specification*, 1.1 edition, May 2001.

[17] HAVi - Home Audio / Video Interoperability. At http://www.havi.org/, 2001.

[18] John Heidemann and Dhaval Shah. Experiences with User-configurable, Location-aware Scheduling. Technical Report 98-675, University of Southern California, April 1998.

[19] IEEE, http://www.ieee.org/. *IEEE Std 1394-1995 IEEE Standard for a High Performance Serial Bus*, 1995.

[20] IEEE, http://www.ieee.org/. *IEEE Std 802.11-1997 Information Technology- telecommunications And Information exchange Between Systems-Local And Metropolitan Area Networks-specific Requirements-part 11: Wireless Lan Medium Access Control (MAC) And Physical Layer (PHY) Specifications*, November 1997.

[21] IEEE, http://www.ieee.org/. *Supplement To IEEE Standard For Information Technology- Telecommunications And Information Exchange Between Systems- Local And Metropolitan Area Networks- Specific Requirements- Part 11: Wireless LAN Medium Access Control (MAC) And Physical Layer (PHY) Specifications: Higher-speed Physical Layer Extension In The 2.4 GHz Band*, January 1999.

[22] All About i-mode. From http://www.nttdocomo.com/i/index.html, 2001.

[23] The unofficial independent imode FAQ. From http://www.eurotechnology.com/imode/faq.html, 2001.

[24] J.N.Weatherall. The r2 low-power messaging and rendezvous layer. In *SIGOPS European Workshop*, September 2000.

[25] Alan Jones and Andy Hopper. The prototype embedded network (PEN). *Computer Networks*, 35:377–390, 2001.

[26] GO project MANET. At http://go.cs.hut.fi/technology/manet.html.

[27] P.V. Mockapetris. Domain Names - concepts and facilities. RFC 1034, RFC 1035, November 1987.

[28] Robert Morris, John Jannotti, Frans Kaashoek, Jinyang Li, and Douglas Decouto. CarNet: A Scalable Ad Hoc Wireless Network System. In Paulo Guedes, editor, *Proceedings of the Ninth ACM SIGOPS European Workshop*, pages 61–65. ACM, September 2000.

[29] Robert Dunbar Poor. HYPHOS: A Self-Organizing, Wireless Network. Master's thesis, Massachusetts Institute of Technology, June 1997.

[30] Jan M. Rabaey, M. Josie Ammer, Julio L. da Silva Jr., Danny Patel, and Shad Roundy. PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking. *Wireless Computing - p42-48*, July 2000.

[31] Philips Semiconductors - Wireless Connectivity - RF-Lite. At `http://www.semiconductors.philips.com/wicon/rflite/`, 2001.

[32] Tristan Richardson, Frazer Bennett, Glenford Mapp, and Andy Hopper. Teleporting in an X Window System Environment. IEEE Personal Communications Magazine, Vol. 1, No. 3, pp 6-12, Fourth Quarter, 1994.

[33] Frank Stajano and Ross Anderson. The Ressurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In B. Christianson, B.Crispo, and M.Roe, editors, *Security Protocols. 7th International Workshop Proceedings*, Lecture Notes in Computer Science. Springer-Verlag Berlin Heidelberg, 1999.

[34] *Universal Serial Bus Specification*, 2.0 edition, April 2000.

[35] Roy Want, Andy Hopper, Veronica Falcao, and Jonathon Gibbons. The Active Badge Location System. ACM Transactions on Information Systems, Vol. 10, No.1, pp 91-102, January 1992.

[36] Roy Want, Bill N. Schilit, Norman I. Adams, Rich Gold, Karin Peterson, David Goldberg, John R. Ellis, and Mark Weiser. The ParcTab Ubiquitous Computing Experiment. *Mobile Computing*, pages 45–102, 1995.

[37] Wireless Application Protocol Architecture Specification. From `http://www.wapforum.org/`, July 2001.

[38] WaveLAN. At `http://www.wavelan.com/`.

[39] James Weatherall and Andy Hopper. Predator: A distributed location service and example applications. In *Cooperative Buildings*, oct 1999.

[40] Wireless Integrated Network Sensors. At `http://www.janet.ucla.edu/WINS/`, February 1999.