Sketching in Software and Hardware Bluetooth as a Design Material

Petra Sundström
Mobile Life @ SICS
Isafjordgatan 22, 164 29 Kista
Sweden
petra@sics.se
(while at Microsoft Research Cambridge)

Alex S. Taylor and Kenton O'Hara
Microsoft Research Cambridge
7 JJ Thomson Avenue
Cambridge, CB3 0FB, UK
{ast, v-keohar}@microsoft.com

ABSTRACT

In any design process, a medium's properties need to be considered. This is generally well established, yet still within interactive systems design, the properties of a technological medium are often glossed over. That is, technologies are often black-boxed without much thought given to how their distinctive material properties open up the design space. In this paper, we experiment with a technology to see what might be gained from intentionally and systematically investigating its properties. Specifically, we look upon Bluetooth from the perspective of being a design material and examine how its properties from that perspective can be used to shape design thinking. Using four example cases or "sketches", we show that Bluetooth's properties, often seen as constraints, can provide useful building blocks for designing interactive systems.

Author Keywords

Design, Bluetooth, The digital material, Design materials, Material properties, Inspirational Bits

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Design

INTRODUCTION

Through sketches, mock-ups and early prototyping, designers engage in a "*conversation with their materials"* [22]. In the formation of new ideas the materials start to "*talk back"*, so to speak, revealing opportunities and challenges. Digital materials—both hardware and software—are though complicated for many designers to work with [12]. They alter both across time and space [9]. Thus, it is not enough to touch and feel digital materials at any given moment to grasp its properties and potentials; instead they only appear to reveal themselves and their dynamic qualities when put to use and, more often than not, assembled as part of running systems. In order to help those who design with digital technologies, we thus need to consider how we might expose the dynamic qualities of digital materials in systematic ways, and critically, ways that make sense to designers, HCI-experts and other members of multidisciplinary interactive design teams.

With this in mind, and as a first step in this direction, we in a previous paper of ours introduced the *Inspirational Bits* approach as a way for engineers and developers to "*open up the design space"* by experimenting with digital materials [24]. In this approach, an inspirational bit is seen as a lose example of a specific technology that allows a multidisciplinary design team to *look* at it, *feel* it and *experience* it over time and space, exposing all or some of the properties of the technology in an inspirational way. From this perspective, it is not assumed that each material has a predefined set of bits. Nor is it assumed that developers and engineers, having not previously worked with a technology, know exactly what bits they will build/find. Instead the Inspirational Bits approach is best viewed as an open-ended exploration of digital materials and their properties. This exploration may be used either: (a) as the first step in a design process (as in a technology-driven design process or a form of *grounded innovation* [15]), or (b) to inform a design team about the properties of the material/s that in parallel with the formation of an idea is the required or most appropriate material/s to be used.

Such an open-ended, playful exploration of technology might be contrasted with more traditional engineering approaches to design. Usually, engineers and developers are taught to find the best solution to a given problem. It is much less typical for them to work on a set of potential solutions at the same time, simultaneously exploring a range of problems and solutions within a particular design space [16]. Thus, our aim with the Inspirational Bits approach is to help enable these opportunities for exploration through a conversation with the digital materials. In this respect the approach has much in common with design *sketching* that both Buxton and Fallman previously have described [4, 6] where there is a gradual and iterative refinement of a design through the testing out of ideas.

MobileHCI 2011, Aug 30–Sept 2, 2011, Stockholm, Sweden.

Having in our previous paper described the Inspirational Bits approach in general, we will in this paper present the details of work on Bluetooth specifically. This as an example case for how to practically go about one of these open-ended explorations of one of the digital materials. And as an example case for how to practically go about exploring one of the digital materials and from such activity find what potentially could be inspirational bits for that material.

The technology, Bluetooth, was for this work chosen for a number of reasons. Broadly, we were attracted to the ubiquity of Bluetooth and its status as a standard for wireless, short-range data communication. We felt this provided us with a technology that is often seen as a closed system or black box with numerous taken for granted properties. Again, the intention was not to solve a specific problem using Bluetooth or to achieve some predefined endpoint. It was rather to uncover and develop a better understanding of the properties of Bluetooth and to see whether a focused investigation of the technology as a design material might open us up to anything different and/or unexpected.

BLUETOOTH

To set some context for the later discussion, we first want to describe some details of the technology. Bluetooth is an open wireless protocol for exchanging data over short distances from fixed and mobile devices. It uses the microwave radio frequency in the 2.4 GHz to 2.4835 GHz range. The radio mode is a broadcasting mode allowing communication between devices when not in line of sight of each other. The range over which Bluetooth can communicate between devices is between 10-100 meters, but typically devices are only a few meters apart when connected (due to walls, furniture, bodies etc.). A master Bluetooth device can connect to and simultaneously communicate with up to seven other devices, called slaves. Such a network group is called a piconet. At any given time, data can be transferred between the master and a slave, however, the devices can switch roles and a slave can become the master at any time.

Bluetooth is either positioned somewhere or carried by someone. Each Bluetooth device has a changeable name and a unique identifier, a MAC address. In actual usage, the Bluetooth chip is embedded in a hardware device or system, such as a mobile phone, laptop, printer or headset. An especially common use of Bluetooth is the mobile phone. A mobile phone equipped with Bluetooth can search for and connect to other nearby devices such as other phones, wireless headsets and PCs. A mobile phone user can however choose between having Bluetooth turned on or off or set it to being non-discoverable to non-familiar devices. Many phone users turn off Bluetooth to save battery-life but a significant few keep it on [19, 21]

When scanning for nearby devices one can, without “pairing”, get access to the name of a device, its MAC address, its device class, the services offered as well as a few other technical details. The name is a text string that can be up to 248 characters long, which can be set to anything the mobile phone owner chooses. While this name is typically set to manufacturer and model, users will often change it in ways that reflect their identities, communicate a message or provoke curiosity [10,19]

For two-way Bluetooth communication to take place pairing is required. However, information can also be pushed to mobile phones via Bluetooth, using, for example, the vCard standard. A vCard is a small text string most often used to pass contact information to another person’s phone. The practice of sending short, unsolicited messages using the vCard protocol is referred to as bluejacking [e.g. 25]. To be stored on a recipient’s phone these messages need to be approved but on most mobile phone models they are, without approval, shown on top of the display while waiting for such approval.

In the following, and with an eye on design, we take a more detailed look at these underlying features of Bluetooth, such as the particularities of how pairing is done, how master and slave roles get switched, the way scanning is achieved and the way data is communicated etc. By looking at the detailed protocols of these mechanisms, the particular constraints, and the kinds of information available for use or manipulation at particular points in the process, we aim to reveal ways that a deeper examination of Bluetooth technology can work as a design resource. We do this through a presentation and examination of some Bluetooth-based design sketches. Again, in presenting these design ideas/sketches/bits, our concerns are not simply with the ideas in themselves but also with what they reveal about Bluetooth that may serve as a resource for designers working on mobile systems. To foreground our primary concerns with the approach, we have chosen to first present our sketches before a more general discussion of how Bluetooth has been approached by other researchers and designers. In doing this, we will also address how the research field might benefit from richer presentations of technologies as materials.

SKETCHING IN SOFTWARE AND HARDWARE

To help extend how Bluetooth technology might be used in new innovative ways or to improve a poorly performing set-up, one has to “*get to know*” the details of the lower level material properties. We discuss what might be characterised as *sketching* in software and hardware. As we have been arguing thus far, the purpose of this software and hardware based design sketching is undertaken not towards a specific goal per se, but with the intention of *un-blackboxing* Bluetooth and its properties. The aim here, then, is not to produce perfectly designed systems, but fully working “sketches” or inspirational bits, that embody some of these properties helpful to us as well as to others interested in mobile interaction systems design and the Bluetooth material. In contrast to some related work we discuss later, we present these designs in ways that more explicitly highlight the underlying mechanical properties of

Bluetooth and how these are exploited in realising the system designs.

We decided to start exploring Bluetooth in the packaging of mobile phones since this is probably the most commonly known package that Bluetooth comes in and also the most widespread. Similarly, we decided to use Java[1] as our programming language. In our last sketch we also work with the Bluetooth Arduino board. All these choices of course affect the characteristics of the material properties we wish to draw attention to. However, as we will see, what we have been working with are very much the basic properties of Bluetooth and therefore this paper should offer value for anyone working in this material whatever the packaging.

BTScores: a first feel

Our first discussions around Bluetooth and what we find to be so interesting with this technology were about its spread usage. Almost everyone through their mobile phone and also many stationary and mobile objects carry Bluetooth and as previously said surprisingly many of those also has it turned on. To set up communication with these other devices is one thing, but first we were interested in what we could do with this design space just as it is. To get a first feel for this we built a system that simply let us collect data from encountered devices. One of the things we discovered that was possible to collect was the *device class*, signified by a number assigned to specific Bluetooth packages encountered, e.g. smart phone, laptop, printer, etc. In addition, it was apparent that what might be regarded as more "rarely" encountered devices are classified with higher numbers. So, for example, a smart phone is defined as 512 while a Bluetooth Arduino board is 7936. From these underlying mechanical properties of the Bluetooth protocol, we got the very simple idea for a points-based game. Building on this property, our development of a game was then further inspired by how children sometimes pass the time in the back seat of a car by counting things like cars and cows and have some sort of point system for that. Our idea for a game was then based on Bluetooth's inbuilt ability to detect other Bluetooth devices around and distinguish their device class. We saw the possibility for a game where we/designers/users would have to run around and "hunt" Bluetooth devices.

We are aware of that the focus here became a bit blurred; in one way we were exploring the technology and in another we were already thinking of what this could do for users and how we could use this to communicate with a design team. However, having tried this also with other technologies we see no other way for this to be done – by necessity it is a messy process. One can think of the processes of uncovering materials as very structured, whereby a designer/engineer simply thinks of and builds one inspirational bit at a time. And also that the first bits/sketches that appear to him or her are the most simple bits and that they, throughout the process, become more complex. Our impression, however, is that it needs to be and also should be quite the opposite. In fact, it is from our experience, also having worked with other technologies than Bluetooth, from building these more playful and perhaps more "designed" bits that we begin to understand the material better and begin to pin down the more basic properties of the material. For this to happen, though, it is essential that the process is to be kept exploratory and open-ended. Again, it is not that there is a specific set of bits to be found for each technology. What bits are discovered depends on who in the design team participates in the process, the potential limitations/directions, previous experiences and more. In these terms, it is most beneficial if the process can be unconstrained for a short period of time. These bits as sketches can always be picked apart later on and not all bits/sketches need to be used/presented to the other members of the design team. Most important is that *someone* in the design team gets to develop a deeper knowledge of the material, or perhaps expands their previous knowledge of that material, and is thereby able to better communicate some of this knowledge to the rest of the team.

Returning to our Bluetooth game, *BTScores* (see Figure 1.), on top of the idea of hunting Bluetooth points (based on device class number) we also added an additional scoring component based on an active pairing with scanned devices. When the game begins, it automatically pushes a message to encountered devices saying "Do you wanna pair with me?". If another Bluetooth user accepts that message, the user playing BTScores gets an extra 1000 points above the points allocated from detected devices. In assessing this game, again, it is important to remember its status as a sketch rather than a complete system design. The intention was to sketch to help understand and articulate ideas to get a better feel for the Bluetooth material. So details about, for example, whether this game might be played on a time limit or in some other context was not relevant to our purposes at this point. As a sketch, and as a potential inspirational bit, the graphical design of BTScore was also intentionally left very poor.

To get a sense of how these particular features of Bluetooth

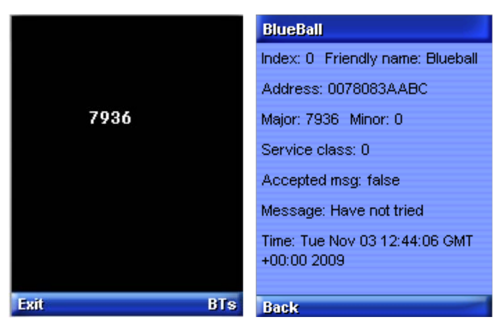

Figure 1. BTScores; the main window and a window holding more details on a specific device, here an Arduino board named the BlueBall

[1] The SonyEricsson version of Java Microedition

might work as potential game mechanics, we played the game out over the course of a few weeks, as we encountered different types of contexts that made us curious about how it would feel when our device automatically Bluejacked someone there while not being able to move away, e.g., when out at a restaurant, or on the bus to the airport. We had chosen to add these automatically-sent messages to encourage ourselves to speak to people about Bluetooth and what it potentially could be used for. It turned out to more be an interesting game of trying to figure out who was using Bluetooth and who was not and to whom an encountered device belonged. Most often it became a game of hiding and trying *not* to be discovered and instead watching people from a distance when receiving our message, (cf Bluejackers). We see potential here for games of more and less complexity based on the excitement of appropriating other Bluetooth users either as non-playing characters, as well as in set ups where the boundaries between playing and non-playing characters are more ambiguous. BTScore is a very simple example of the software and hardware sketching practice but nevertheless is a useful starting point for the discussion. We find additional richness in our second example, *BluePete*.

BluePete: in-visible active and visible passive

When setting up communication between devices, as in the additional scoring component described above, we started to realise more of what it actually means for a Bluetooth device that it cannot search and listen at the same time. How a Bluetooth device cannot be slave and master at the same time. How the Bluetooth protocol is implemented so that a device searching for other devices to connect to, only will find *passively-listening* devices and not devices that also *actively* are searching. It is, though, possible for a device to switch between these two *listening* and *searching* modes. This switching possibility inspired us to think about how something would be swapped or exchanged by changing identity. Based on this, we decided to create a Bluetooth game based on the Swedish card game called "Svarte Petter", where the game is to not have "Svarte Petter" when the game is over. In the card game version a special deck of cards is divided between the players and the procedure is then to take cards from each other. As players collect pairs of cards, these are discarded. There is only one "Svarte Petter" in the deck and when only that card is in game the game is over and the person holding this card looses (the winner is the one that first has no cards left). In our Bluetooth version of this game we take advantage of the particular characteristics of the searching protocol discussed above, whereby a Bluetooth device cannot *search* and *listen* at the same time. The *BluePete* game (see Figure 2) is a simplified version of Svarte Petter and does not involve a full set of cards. Rather, the game is simply one of having BluePete or not. When your device has BluePete, the device searches for other devices running the game to which he can be offloaded. When finding a device to push BluePete to, the device switches from searching to listening. By doing this, it opens itself up for the potential of getting BluePete again. Since no device can search and listen at the same time it can be assumed that devices found do not have BluePete, which means that very little game control needs to be handled when two devices find each other. If two devices find each other, that means that one is *listening* and one is *searching*, In terms of the BluePete game, this essentially means one device has him and the other device does not. Thus, we can see how different states available in the underlying Bluetooth protocol can be utilised as the basis for new game mechanics when set within an appropriate game narrative.

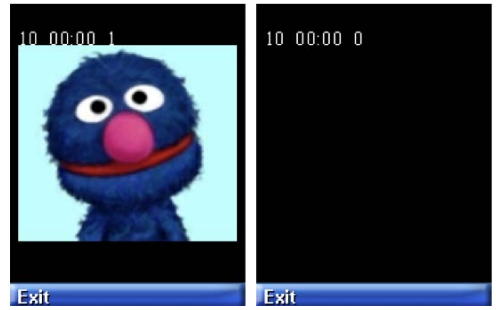

Figure 2. BluePete; on the left showing that has him and on the right showing a user that has never have had him

Further, an important aspect of this searching behaviour is the time taken to perform one of these device scans. This is an important characteristic in thinking about appropriate game mechanics based on these Bluetooth features. Most of the papers on Bluetooth based applications make some reference to this process taking a "long time". For example, Aalto and colleagues calculate that on average, a device spends approx 10 seconds in inquiry mode [1] with it sometimes taking up to a minute [13]. But there is little discussion of the things that affect this time, such as particular device type or environmental factors. Again we would want to argue that these kinds of details can be important and incorporated explicitly as part of the game experience. To describe a time period as a "long time" does not really provide us with any meaningful information on which to build since what is considered long or short is relative to the context of use. What we would argue is rather than describing this period simply as a long time, these time periods need to be understood and used appropriately as a game mechanic in an appropriate game narrative. In BluePete, we can consider this time period as a component of the game and part of the excitement of playing – forcing people to stay proximate for the time period before the potential offloadee runs away. Fighting the material will not present the idea to its best. But, by experimenting with and getting to know the technology we can reveal ways to work on the problems and different ways of doing things that can open up new ideas.

Let us now consider other aspects of the search and connect protocol. In using Bluetooth and Java on a mobile phone a service need to implement the following:

- **Stack initialization** – Initializing Bluetooth
- **Device search** – Starting a search for all devices nearby
- **Device discovery** – One by one those devices report back to the device that initiated the search
- **Service search** – For each device reporting back it is possible to search for specific services (not possible to begin with)
- **Service discovery** – For other devices, if they have the services being sought for, report back to initiating device
- **Connecting** – and if that happens a connection can be set up

It is not possible to search for devices that run a specific service, such as, for example BluePete in the first iteration. BluePete is set up to be an ad-hoc system to be played there and then without a set up server somewhere and by anyone having the system software - therefore there are no known MAC addresses. In an alternative solution, having the address of a known server, a connection can be set up already after having discovered that device, a solution that of course is mush faster but not possible here. To speed up the game and open it up for players actually having to run when having pushed BluePete to someone, the BluePete game keeps a list of all devices to which it has ever pushed BluePete and tries this list first before performing a new device search. We also make sure that we try the last pushed to devices first since these are more likely to be close. Still, this design decision is not perfect in that it makes it harder for new players to become part of the game. What we start to see here then is how design trade-offs are revealed here based on the particularities of the underlying Bluetooth protocols. Rather than just being a concern of technical implementation, then, the details of an un-blackboxed technology can actually be the basis of interesting design choices. Revealing this un-blackboxed detail to the broader design team can be considered an important component of the design process.

Similarly, we can consider what is involved when a device *listens* for searching devices that want to connect to it. This listening procedure is much quicker and less complex but nevertheless a similarly interesting design resource:

- **Stack initialization** – Initializing Bluetooth
- **Service Registration** – Making sure that the device is in discoverable mode and that it presents the right services to devices that tries to connect to it
- **Waiting to be found**

There is also the choice of being in *non-discoverable* mode where only devices that know the specific MAC-address of the device can find it and establish communication. In some cases this might be the best solution for hiding from someone while being there for others who are, for example, more initiated or perhaps team members in some game.

BluePost: someone or some

Through BluePete we become aware of differences between general and specific (when a specific MAC address may be known). Through understanding these differences we began to think how these properties could be utilised for other forms of communication-based mechanics. One such form of communication, is the "message in a bottle" sent out in the world with the hope it might be passed on to someone general or someone specific. With this in mind, we developed the *BluePost* system (see Figure 3). In BluePost you write a message and direct it to some device name, one that you might know of or one that you think is out there somewhere, such as "Nokia 6300" (Kindberg and Jones report on the majority of those interviewed said they know of at least one Bluetooth name of someone else [10]). The idea is then that the user carries this message until s/he either comes in the presence of either another device having that same Bluetooth name or another device also running BluePost. At this point, the message is transferred to be carried by the new user until s/he meets another phone of the same name or another BluePost user – and so on until the message finds its "receiver".

Changing from searching for a specific device with a specific MAC address to instead searching for a device having a name equal to the one set as the receiver of some message in BluePost also opens up for a message having multiple potential receivers. The messages in BluePost are not copied between pairing devices but pushed. This means that only one of those potential receivers could actually get it. This difference between searching for a unique MAC address and searching for a name that potentially several devices could have is another interesting property of the Bluetooth material that we want to highlight.

But this idea also brings us back to the problematic issue of the Bluetooth material not being able to search and listen at the same time and by that being invisible when searching. Carrying a message in BluePost means a client needs to both listen for potential messages coming in as well as searching for receivers or other carriers. This needs to be achieved in as short intervals as possible since devices potentially move passed each other quite quickly. Initially, we worked with our BluePost idea to see how far we could push it in terms of switching between searching and listening modes. Keeping in mind that two devices might end up in the same intervals and thereby never hear each other (due to searching and listening at the same time) we first tried using randomly long intervals. We checked further with more experienced software programmers at SonyEricsson to see if they had a solution to this. We were

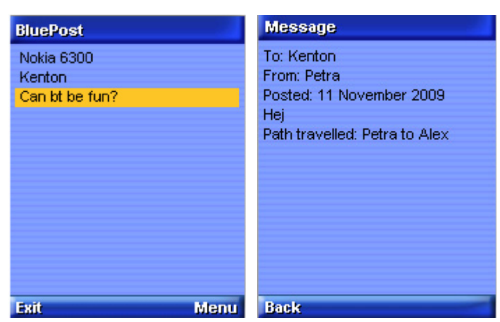

Figure 3. BluePost; first the main window holding all carried messages then a message carried for Kenton (the user is Alex)

referred to the Bluetooth specification for the specific phones we where aiming for, but while finding information on the Bluetooth technology being able to switch between the two modes there was nothing on how and in what speed etc. Basically, then, the Bluetooth technology is designed to be used *either* as a searching device or a listening device. At this point we found ourselves fighting the material. However, we were reluctant to give up completely on the BluePost idea. As such, we discussed the possibility of using two Bluetooth chips, one for listening and one for searching. While, on the face of it, this appeared an "ugly" solution from an engineering perspective, we came to realise it was an interesting idea for the problem at hand. Indeed it made us consider this as a more interesting general idea beyond the specific solution at hand, opening up potentially new design opportunities for Bluetooth application not available with the traditional single chip approach. In a design team where different people are skilled on different things it might also be that these solutions are so basic or so "not right" to some that they are not discussed or considered. Important here, then, is by understanding more of the material mindset of each discipline (designers, programmers, etc), we might achieve a larger solution set to choose from.

BlueBall: reversed control

Related to the issues of searching and listening are the *master* and *slave* properties of Bluetooth. Let us consider, now, some of the material properties of this master-slave relationship with a view to inspiring further design ideas. A master device, e.g., a PC, initiates the connection to a slave. It can connect to seven slaves at once, this connected cluster called a *piconet*. A slave, however, cannot establish any connections; instead, it acts as a listener to incoming connections from the master device. Thinking about this in terms of the client-server paradigm, the Bluetooth master device behaves like a client (like a client, initiates the connection to the server), whereas the slave behaves like a server (i.e. listens for incoming connection requests). In short, the master is a client; the slave is a server. Yet the master device is capable of connecting to multiple slaves and holding these active connections simultaneously. So in other ways, it behaves like a server, like a web server, for example, it can serve multiple client browsers. In this sense, the master behaves as a server and the slave as a client. The apparent roles of Bluetooth masters and slaves, then, alter under different conditions.

The slave-master relationship can also be considered in terms of the roles of control or being controlled. With Bluetooth *one* pairing has *one* connection – thus, there is no broadcast and therefore poor point-to-multipoint support. So, from a technical perspective, one client (master) creates connections to seven servers (slaves). But, from the application point-of-view, the master also has to have the intelligence to communicate or echo the events from the slaves to each other; all traffic goes through the master (an arrangement similar to a traditional game server).

These technical details of the master-slave relationship in Bluetooth, led us to something we have called "reversed control". Inspired by the piconet property, we wanted to create a set-up in which an object might be controlled collaboratively by a group of Bluetooth users, e.g. trying to steer "*BlueBall*" (see Figure 4). To achieve this, we considered a simple sharing of the controls. For example, with five people in a group, one would control *move right*, a second, *move left*, a third, *move back*, a fourth, *move forward* and a fifth, *stop*. The group would then cooperate to steer the ball. In this solution, the ball is the master and can initiate connections with up to five users. However, the difficulty here is that since the master also initiates the connection and is not an idle actor, the arrangement is not viable. An alternative solution considered was to have the BlueBall look for mobile phones equipped with a specific client. However, this leaves the problematic mobile phone set up where the differences between various phones become yet another material that should perhaps be examined on its own. The eventual solution we chose was for the Ball to scan its surroundings and to define the various Bluetooth devices it finds to steer it. To steer the ball, connected users figure out which Bluetooth devices are associated with the different directions. They then move the device in and out of Bluetooth range to steer the ball. While this idea is somewhat unusual, we believe it opens up an interesting and playful approach to indirectly controlling something like a ball.

On the face of it, what we aim to convey here with our Bluetooth sketches/bits, may seem familiar and unremarkable. Yet, what strikes us is how, on a regular basis, this particular role for the detailed underpinnings of technology as a design resource has not really pursued with deliberate intent. Indeed, the default position has often been to think of the technology as a means to solve a defined problem. It is only when we start getting our hands dirty, so to speak, that we have found ourselves tweaking the underpinnings of a technology and then hopefully arriving at some compelling possibility. Seldom has such an exercise been pursued as a systematic component where it is deliberately used as a resource for expanding or *opening up the play of possibilities for design* (to borrow a phase from another context [2]). In taking this stance, it is important to distinguish the approach from traditional

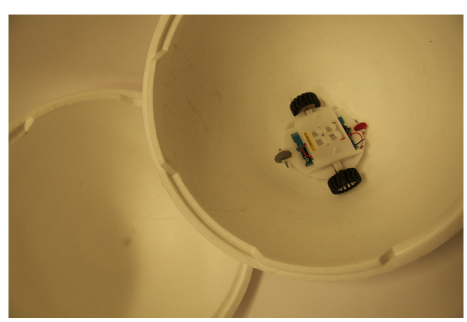

Figure 4. The BlueBall flipped open revealing the Arduino Bluetooth master device.

techno-centric innovation. The approach we are discussing remains closely tied to user and design concerns. However, in bringing together user, design and technological concerns, the approach we explore in this paper argues that we can start from different points in this space – and in this particular instance, from the material and architectural properties of the technology [cf. 8, 7].

As we pointed out in the introduction, we have from these sketches presented here (and from sketches we have built in other materials) started to formulate our idea of inspirational bits as a way for multidisciplinary design teams to work together with the digital material as a design material. What is further necessary here, and an endeavour that we have started [24], is a continued and systematic evaluation of these sketches and others from the perspective of them being/becoming inspirational bits. In this paper we wanted to communicate the details of our explorations of Bluetooth as an additional clarification of how we believe this process of uncovering materials needs to be both playful and open-ended - and to further highlight the values of making the digital material a more prominent part of the design process in shared forms from which all members of interactive design teams can take inspiration.

REVISITING THE DESIGN SPACE

It should be said that we are of course not alone in addressing the different mechanical properties of Bluetooth in interesting ways. So, before concluding this paper, we want to first discuss some projects that have used Bluetooth as building blocks in an interactive system design. In choosing these particular projects, the aim is to highlight some of the explicitly interesting building blocks that have been used in the ways we envisage but also draw attention to building blocks of interest that are not explicitly articulated in the work. This latter point is particularly pertinent for our current purposes in that very few of the papers or web sites describing these systems articulate more than a system description or a design idea in general. Typically, the focus of these system descriptions means that the details of how Bluetooth has inspired a design or how the Bluetooth building blocks have been used are not explicitly articulated. Some do mention technical problems, such as difficulties with devices actually finding each other, but there is very little explanation on why these problems arise and the effects of an alternative set up. It is these details that we want to draw more explicit attention to and show how they can be a source of inspiration in design.

Many of the researchers on these projects will be familiar with the difficulties in working with Bluetooth and certainly a significant proportion of the papers referring to Bluetooth, published in HCI-related fields, suggest it is common to adapt Bluetooth, altering the intended purpose of the technology as a file exchange mechanism. For example, they may use Bluetooth to simply transfer metadata, such as information about the songs others nearby are listening, e.g., BluetunA [3]. This is possibly because Bluetooth is in some ways is quite poor as a file-exchange protocol, in that, as we have seen, it can take some time to connect two devices and transfer larger data files. Most designers working with Bluetooth seem, then, to be more inspired by the simple fact that a Bluetooth device can discover names of nearby devices or access superficial data without pairing.

Systems making use of nearby devices

Paulos and Goodman [20], talk of the "familiar stranger"; people who while not formally acquainted, see each other regularly in the same physical space by virtue of travelling the same way everyday (e.g. due to commuting to and from work) and thus start to recognize each other. In their system, Jabberwocky, Paulos and Goodman present the names of all the Bluetooth devices a user passes in his/her daily life. Some of the names users might start to recognize and also even connect to a person or a place they regularly pass. Along a similar vein, users of Cityware can in their Facebook account follow the statistics of the people they meet and the places they pass [11]. In another system, MobiTip, [21] Rudström and colleagues have also wanted to provide users with a feel for the people and places they regularly pass. In their system they let users rate and comment on things they do and see. These ratings and comments can be shared between devices using Bluetooth, providing users with recommendations about where they are. The overall idea is that when two people pass each other, one user's past might become the other user's future.

All of the above systems are excellent innovations but none of their descriptions mention ways that prevent Bluetooth technology being found when searching. For example, if there are two or more users of Jabberwocky or Cityware in the same place, many of those users' devices will all be searching at the same time. The consequence of this is that devices simultaneous searching for each other will actually not be able to find each other. Rather, it is only those devices that are *passively* listening that will be found by devices *actively* searching. In the MobiTip and BluetunA systems, this becomes especially problematic in that users of those systems are meant to exchange information with each other. For that to happen devices running those systems must find each other. How that is technically solved is not presented in the papers we have found on these systems. In personal communications with the people behind the MobiTip system about this, they acknowledged this as a problem. As a workaround, they used an algorithm whereby the device would search for 20% of the time, and less when fewer and fewer devices were found in the device scans. But this still meant that there were many times where devices passing each other were both listening and therefore not finding each other. There must also have been occasions were two devices both were searching although not as often.

In some senses, we might regard this as a very technical issue and approach it as something to be solved or worked around. Our argument, though, is that rather than just seeing this as an inherent technical problem and therefore not worthy of explicit articulation, we can also approach this particular characteristic as a potential design resource. In this respect, we would argue that such a technical setting of not being found when searching, is something we can draw attention to as a design resource and therefore deem it worthy of more explicit articulation.

More narrative usage of nearby devices

Another interesting use of Bluetooth's basic properties as a design feature can be found in the work of Lautamaki and Suomela [13]. In their Sandman game users can either sprinkle sand to make other users sleep or drink coffee to stay awake and protect themselves from the sand other users might sprinkle. However, again, like many other papers published in HCI, while this is a compelling system that exploits the properties of Bluetooth in novel ways, the building blocks of the system are not described in sufficient detail for others to take inspiration for further design. As such we have to make certain assumptions of these details from our own experiences of Bluetooth that are not available to others who do not have this experience. This is especially frustrating here though since the Sandman game seems to be very much in the direction of the more innovative use of technologies that we are looking for. We can speculate the particular game characteristics could be technically achieved by putting a device in non-discoverable mode for a while and therefore not being found by a device searching for other devices to sprinkle sand on. It might also be implemented using a Boolean variable for drinking coffee – it is unclear. From both a user perspective and technical perspective this is an important difference. Not only does the later require more time in that more devices will be found in a device scan, but also there need to be communication with all devices found to check for such variable.

Another innovative game using properties of the Bluetooth technology in an innovative and interesting way is the HotPotato game built by Niemi and colleagues [18]. The aim of the game is to collect points by trying to hold on to a set of potatoes that get warmer and warmer. If they get too hot a user needs to find a place to let them cool off and later return to that place and bring them back again. Niemi and colleagues talk about *non-playing characters* - people who don't run the software client and who are unaware that the presence of their Bluetooth device is being used as a game feature. In the game, these *non-playing characters* can be used by active players, as places where they can cool off potatoes that are too hot. In technical terms, the game is to find another MAC address and use that as the signifying key in a database; nothing is actually stored on the non-playing characters' Bluetooth devices. The user of the other device is never notified of their MAC address being used like this - the HotPotato game exploits the technical setting in the Bluetooth technology that allows the MAC address to be obtained without the user of another device being notified. The twist here is that a user needs to find that same other MAC address again when s/he later wants her potato back. If that MAC address belongs to a person and not a place it can be quite complicated and the user might need to follow strangers around. Again, how this is technically achieved, using the MAC addresses of nearby devices, is our assumption from reading the game description and from our own experiences with this technology rather than something made explicit by the authors. Linehan et al [14] have read the paper from the same perspective as us and used this same technical set up in their innovative and somewhat controversial pervasive game where they let non-playing characters smuggle virtual drugs through real airport security.

Another interesting detail of the HotPotato game, that takes us back to how two searching devices cannot find each other, is when a *pushed-to-user* is actually another player. In this instance, the description of this game tells us that the other user has the choice of throwing a potato back and by that burn the initiating user. In this game we do not see the same problem as previously discussed with reference to Jabberwocky and MobiTip. Searching for other devices to push potatoes to is, in the HotPotato game, not a constant state. There will be some devices searching and some waiting for their potatoes to warm up.

Spanek and colleagues, in their paper on the BlueGame project explicitly state that they have made use of and been inspired by the fact that the Bluetooth technology can be used to form piconets [23]. That is, how one device can take the role as master and at the same time hold communication with seven slaves. This paper is though only two pages long and does not really explain this beyond the fact they have used this Bluetooth property as the basis of a multi-player game based on Dungeons&Dragons.

Proximity-based or Location-based usage

A further material property of Bluetooth used in innovative system design concerns Bluetooth as a short-range radio technology. In such usage, if a device is found in a device scan it is considered as being *nearby* by virtue of Bluetooth's operational range. For example, the Cityware system mentioned above, uses closeness to prepositioned stationary Bluetooth devices to determine the location of users. Another system, B-MAD delivers permission-based location-aware advertisements to mobile phones using Bluetooth beacons distributed around the pedestrian streets of central Oulu in northern Finland [1]. In B-MAD people are not positioned as in Cityware. Rather, here it is simply a correctly placed beacon that broadcasts these advertisements and by default only devices nearby will get them. There are also a few more commercially developed systems using this Bluetooth property of being proximate to be found. While conceptually, these systems are of interest, there is some uncertainty as to the extent to which these systems may effectively work. For example, in some typical scenarios of use, these systems would arguably require much faster connectivity time and faster data transfer than is realistic from our own experiences with these factors. Similarly, Marketeye (www.accinity.de/www/en/marketeye/) uses Bluetooth technology to deliver videos, images, music, games or shopping vouchers to nearby mobile phones. Marketeye is the server device sending these file packages and can be placed as a physical device in shop windows, neon signs or even promotion cars. We can see that this probably is a working solution waiting for the bus in front of a neon sign where users stay in the proximity of the device for quite some time. But it is unrealistic to assume this might work between users and a passing car. This also since it is more a rule than not that a Bluetooth device will find all Bluetooth devices in one single scan. And yes, connectivity times and the time it takes to complete a data transfer are issues of the Bluetooth material that probably will continue to improve. But our point here though, is also that these properties of the Bluetooth material as they are can be used in interesting and innovative ways. For some ideas it might even be a feature that the Bluetooth technology has a rather slow rhythm to it, as in the BluePete scenario.

Furthermore what we also want to put focus on here is that in most cases it is the problem that drives the solution. What the B-MAD and the Marketeye system most prominently is all about is to position people and get some data across, for that Bluetooth might be a good enough solution but there might also be other and better suited technologies to accomplish this. What we want to open up is for the problem and the solution to drive each other - explore a technology to get inspired.

Davies and colleagues have made intriguing use of the Bluetooth naming property in combination with the above material property of closeness. They let users interact with an e-Campus system by changing the name of their device into one of the following commands: map, flickr, youtube, google, tiny and juke [5]. To sort out these names from other Bluetooth names found in a device scan each command needs to start with ec for e-Campus system. A command can also be followed by a further defining search command. A user changing the name of his/her Bluetooth device to "ec google elvis" in front of one of the publicly located computer screens in this campus system will then affect one of these screens to present the result of a google search on the name Elvis.

Maunder and colleagues have similarly to us been interested in a deeper understanding of the Bluetooth technology and thereby been able to develop the Snap and Grab method for communication without an exchange of a pass key and without running a specialist software client [17]. The basics of this method is that users take a picture of the media they want and send that photograph over Bluetooth to a public display providing this media. The public display remembers the MAC address of that picture and is thereby able to send back the data that corresponds to that photograph without having to pair with the requesting device.

In this last section it is not so much the issue that the building blocks are not exposed but that they are not presented as building blocks to a larger design community. Technically clever solutions such as the Snap and Grab method are foremost presented to a more technically oriented community and might never reach a more design-oriented community.

CONCLUSIONS

Our intention with this paper has been to demonstrate how Bluetooth can be approached as a design material. What we tried to do in each example sketch was to begin from some of the properties of Bluetooth and subsequently use what we discovered to explore a set of design possibilities. Crucially, we avoided viewing technical specifications of Bluetooth as constraints to be overcome and, instead, treated them as starting points to open up design spaces.

Through this work, we have identified several properties we feel valuable in this respect. These can be summarised as follows:

- **Carried or Placed** - A Bluetooth chip is either carried by someone or placed somewhere, which in turn opens up for interaction with people, places and/or things nearby.
- **Changeable Name** – that can be changed into a command, to show belonging to a group or in various other ways.
- **Unique Identifier** - which can be used when wanting to treat all users and places as unique.
- **Embedded** – The hardware or system device Bluetooth comes in on the one hand brings its own set of interesting material properties and on the other hand provides a class defining number that tells what kind of Bluetooth device that is encountered.
- **Search or Listen** – A Bluetooth device can not search and listen at the same time and when actively searching a device can not be found (also two listening devices will not find each other).
- **Non-Discoverable Mode** – This is the third mode a Bluetooth device can be set to, which means that only familiar devices will be able to find it, devices that already knows its MAC address.
- **Short-Range Radio Technology** – Only people and places nearby will be found in a device scan.
- **Exchange of Data** – Two Bluetooth devices can exchange data and files with each other, more takes longer time.
- **One is Not Enough** – Not all devices will be found in one single device scan. A device will have to be set to repeated scans in order to find all devices.
- **Master and Slave Properties** – A master device can connect to and hold communication simultaneously with up to seven passive slave devices (such group is called a piconet).

Many familiar with Bluetooth will, of course find some of these properties familiar. It is also likely that some of the listed items could be aggregated by looking through past publications (some of which we have cited) and reading the technical documentation on Bluetooth. However, as we have argued above, in many cases, these properties are not explicitly discussed or, alternatively, they are treated as resources to enable problem-led design solutions. In short, they are largely treated as the technical details that must be dealt with in building an envisaged system.

We began this paper by suggesting it was just this kind of *blackboxing* of technologies that we wanted to reconsider. We were interested in what we would find if we started the design process by thinking of a technology like Bluetooth as a material that's properties could be exposed and configured in new ways. Over and above our assembled list of Bluetooth properties, then, what we hope our work has provided is an early insight into a design approach where open-ended, exploratory exercises with a technology are intended to produce the formative building blocks of a design idea and thus seen as a valuable, if not critical, part of the process. The Inspirational Bits approach [24] as suggested in here could be one way in which this can be done.

AKNOWLEDGMENTS

The research was done at Microsoft Research in Cambridge and at the Mobile Life centre, funded by VINNOVA, Ericsson, Sony Ericsson, TeliaSonera, Microsoft Research, Nokia and Stockholm City Municipality.

Special thanks to Bo Larsson at SonyEricsson for great many phones. And to Åsa Rudström, part of the MobiTip developing team, and our colleagues at the Mobile Life Centre, for their comments on previous drafts of this text.

REFERENCES

1. Aalto, A. et al. (2004) Bluetooth and WAP Push Based Location-Aware Mobile Advertising System. MobiSys'04, Boston, USA.
2. Anderson, R. J. (1994). Representations and requirements: the value of ethnography in system design. Human-Computer Interaction, 9(1), 151-182.
3. Baumann, S. et al. (2007) BluetunA: Let Your Neighbour Know What Music You Like. CHI'07, San Jose, USA.
4. Buxton, B. (2007) Sketching User Experiences - Getting the Design Right and the Right Design. Morgan Kaufmann.
5. Davies, N. et al. (2009) Using Bluetooth Device Names to Support Interaction in Smart Environments. MobiSys'09, Krakóv, Poland.
6. Fallman, D. (2003) Design-oriented human-computer interaction. CHI'03, Ft. Lauderdale, Florida, USA.
7. Gellersen, H. et al. (2005) UbiPhysics: Designing for physically integrated interaction. Ubicomp'05, Tokyo, Japan.
8. Greenberg, S. and Buxton, B. (2008). Usability evaluation considered harmful (some of the time). CHI'08. Florence, Italy.
9. Hallnäs, L. and Redström, J. (2006). Interaction Design - Foundations, Experiments, The Interactive Institute, The Swedish School of Textiles, and University College of Borås.
10. Kinberg, T. and Jones, T. (2007) "Merolyn the Phone": A Study of Bluetooth Naming Practices. Ubicomp'07, Innsbruck, Austria.
11. Kostakos, V., O'Neill, E. and Jones, S. (2008) Social Networking 2.0. CHI'08, Florence, Italy.
12. Kursat Ozenc, F. et al. (2010) How to support designers in getting hold of the immaterial material of software. Proceedings of the 28th international conference on Human factors in computing systems. Atlanta, Georgia, USA, ACM.
13. Lautamäki, J. and Suomela, R. (2008) Usimg Player Proximity in Mobile Multiplayer Games – Experiences from Sandman. ACE'08, Yokohama, Japan.
14. Linehan, C. et al. (2010). Blowtooth: pervasive gaming in unique and challenging environments. Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems. Atlanta, Georgia, USA, ACM: 2695-2704.
15. Ljungblad, S. and Holmquist, L. E. (2007). Transfer Scenarios: Grounding Innovation with Marginal Practices. CHI 2007, San Jose, CA, USA.
16. Löwgren, J. and Stolterman, E. (2004). Thoughtful Interaction Design: A Design Perspective on Information Technology, The MIT Press.
17. Maunder, A., Marsden, G. and Harper, R. (2007) Creating and sharing multi-media packages using large situated public displays and mobile phones. MobileHCI'07, Singapore.
18. Niemi, J., Sawano, S. and Waern, A. (2005) Involving Non-Players in Pervasive Games. Critical Computing 2005, Aarhus, Denmark.
19. O'Neill, E. et al. (2006) Instrumenting the city: developing methods for observing and understanding the digital cityspace. In: Dourish, P. and Friday, A. (eds.) UbiComp'06. LNCS, vol. 4206, Springer, Heidelberg.
20. Paulos, E. and Goodman, E. (2004) The familiar stranger: anxiety, comfort, and play in public places. CHI'04, Vienna, Austria.
21. Rudström, Å. Höök, K. and Svensson, M. (2005). Social positioning: Designing the Seams between Social, Physical and Digital Space. HCII'05, Las Vegas, USA.
22. Schön, D. A. (1983). The Reflective Practitioner: How professionals think in action. London, Temple Smith.
23. Spanek, R., Kovar, P. and Pirkl, P (2007) The BlueGame Project: Ad-hoc Multilayer Mobile Game with Social Dimension. CoNEXT'07, New York, USA.
24. Sundström, P., Taylor, A. S., Grufberg, K., Wirström, N., Belenguer, J. S., and Lundén, M. (2011). Inspirational Bits - Towards a Shared Understanding of the Digital Material. CHI'11, May 7-11, Vancouver, BC.
25. Thom-Santelli, J., Ainslie, A. and Gay, G. (2007) Location, Location, Location: A Study of Bluejacking Practices. CHI'07, San Jose, USA.