Avoiding "Big Brother" Anxiety with Progressive Self-Management of Ubiquitous Computing Services

Kevin Feeney, Dave Lewis, Kris McGlinn, Declan O'Sullivan
Knowledge and Data Engineering Group, School of Computer Science and Statistics, Trinity College Dublin
{kefeeney|dlewis|d.osullivan|mcglinnk}@cs.tcd.ie

Anne Holohan
Department of Sociology,
Trinity College Dublin
aholohan@tcd.ie

ABSTRACT

Despite the significant research over the last ten years, commercial ubiquitous computing environments and pervasive applications remain thin on the ground. This paper looks at the explosion in application creativity on the internet in recent years – the so-called 'web 2.0' – in order to identify the obstacles to application creativity in ubiquitous computing. Although technological and standardisation advances are progressively diminishing the scale of the technical problems in the domain, how to manage such applications in such a way so as to encourage user-acceptance remains an open question. It is a question that is particularly difficult due to the serious privacy concerns and the need for negotiated management of services between users due to physically embedded nature of sensor-driven applications. We describe a technical platform which is designed to allow users of ubiquitous computing environments to manage their own personal data and share it in a controlled way and describe an experimental programme to measure the relationship between users' perception of how much control they can exercise over their personal data and their acceptance of ubiquitous computing applications.

Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/Machine Systems – *human factors*; H.5.3 [**Information Interfaces and Presentation**]: Group and Organisation Interfaces – *collaborative computing, organizational design, evaluation / methodology*; H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems – *artificial, augmented and virtual realities.*

General Terms

Management, Measurement, Design, Experimentation, Security, Human Factors.[1]

Keywords

Ubiquitous Computing, Simulation, Collaborative Management, Privacy Management, Policy Based Management.

MobiQuitous 2008, July 21-25, 2008, Dublin, Ireland.

1. INTRODUCTION

Although Ubiquitous Computing (UbiComp) environments, which make use of embedded sensors to deliver more intelligent user-services have been a topic of research for over a decade, there have been few serious attempts to develop commercially viable implementations. Successful ubiquitous user-applications are generally limited to particulars devices (e.g. mobile phones, Ipods). This is a consequence of characteristics of UbiComp environments which present particular difficulties for designers and administrators of services. This paper looks at the relative dearth of innovative UbiComp applications and contrasts this with the large amount of creative and innovative applications which have been enabled on the web by "Web 2.0" technological and social advances.

One important barrier that UbiComp developers have faced is the lack of technological stability and a standardised platform compared to what is now available to web-developers. However, advances in mobile device and sensor technology have started to address some of the underlying technical problems. For example, the proliferation of Java Virtual Machines has provided something of a common platform for service developers, while Wi-Fi (802.11x) is becoming much more widely available, even on simple embedded sensor chips. The advent of Open APIs and open-source development methods has also eliminated some of the important barriers to innovation.

While many of the obvious technical barriers are being actively addressed, there remain significant problems which stem from the business, social and organisational impacts of UbiComp capabilities. From a business point of view, the major problem is that there are very few commercially successful examples of UbiComp services. From an organisational point of view, one problem particular to the domain is the fact that applications may make use of devices and networks that are owned and operated by a wide range of different groups and individuals. For example, a UbiComp service may make use of a user's PDA to convey messages derived from the environment to the user. Thus, UbiComp service designers cannot make any general assumptions about the ownership of the various devices which participate in delivering their services to users. Some devices may be personal, others may be embedded in the user's environment and operated by third parties. UbiComp service providers need to deal with environments where the authority over the available resources in any particular setting is diffuse. In terms of traditional network management, UbiComp environments present potentially extreme examples of multi-domain management. Finally, from a social point of view, sensor data provided by UbiComp environments

creates grave privacy concerns for users which are a significant barrier to user uptake.

This paper analyses the problems which have hindered the development and dissemination of innovative UbiComp applications. Based on this analysis, we adopt a basic, high-level approach to tackling these problems. In order to overcome fears about user privacy and encourage the uptake of applications which depend upon rich sensor-derived data, we adopt an approach which aims to put decision-making power about how each user's personal data can be accessed in their own hands – including facilities for delegating these decisions in a controlled way to third parties. Our hypothesis is that if users perceive themselves to have control over their personal data, they will be more likely to trust applications which use that data. Having outlined our basic approach to addressing these management problems, we describe the technical framework that we have implemented to support the distributed, progressive self-management of such services. Finally, we describe a programme of socio-technological experimental research, which we are undertaking in order to further our understanding of the domain.

2. WEB 2.0 AS AN EXAMPLE OF A CREATIVITY-ENABLING PLATFORM

The world of "Web 2.0" provides a good model for where we would like to get to in UbiComp development. On the web, plummeting hardware and bandwidth costs have meant that application hosting services have become commoditised. Operational costs dominate and service providers can rent inexpensive hosting and avoid significant capital investments. The availability of open web-service APIs and the mash-up architectures that they enable, with the increasing importance of user-generated content, has facilitated business models purely based upon advertising revenue and third-party content. The low cost of entry that this model allows, has resulted in an explosion of creativity among application developers in the Web 2.0 world.

One of the major challenges facing the UbiComp field is in encouraging the development of innovative applications and services which make use of UbiComp capabilities. The development of applications which leverage information culled from sensors will be crucial in making UbiComp enabled device purchases and visits to UbiComp environments attractive to users. Creative applications will be beneficial to equipment vendors and commercial smart space operators. But, the question remains, is it possible to provide a development environment for UbiComp environments which could enable light-weight application development in a similar way in which the web's technical infrastructure has evolved to provide many of the tools required for commodity application development?

One of the key enablers in unleashing the creativity of application developers on the web was the increasing maturity, standards-compliance and stability of web-browsing technology. The stability of the underlying technologies provided solid foundations for the development of a wide range of libraries, allowing developers to much more easily build rich-client interfaces without having to re-invent the wheel. In combination with the development of asynchronous 'chunk-oriented' web-server communication paradigms, AJAX client libraries have drastically reduced the cost of developing complex user-oriented applications on the web.

UbiComp applications do not enjoy the same maturity and stability of platforms that modern browsers offer to application developers. However, the increasing availability of Java Virtual Machines on a very wide range of mobile and embedded devices, based on open specifications such as the J2ME standard and the Squawk-based Sun SPOT project, shows that mobile and wireless devices may, sooner rather than later, offer a relatively standard, stable and extensible application development environment. While this standardized development environment for UbiComp applications is emerging, it still falls short of the web's support of lightweight interpreted scripting languages such as PHP. Furthermore, while open APIs for embedded processors are becoming more common, the domain still lacks a service and application inter-operability framework comparable with that available to web 2.0 application developers.

The problem of monetizing UbiComp application development also sets it apart from the Web 2.0 world. It remains unclear as to which income streams will be available for developers. While much of the application-creativity of the web has been ultimately based upon advertising revenue, mobile devices generally lack the display "real-estate" to accommodate significant volumes of advertising. Other revenue models – such as downloaded licences or Software as a Service (SaaS) – are problematic as, while they may work for some mobile devices, it is unclear how they might be applied to environmentally embedded, sensor-dependant applications.

The lack of clear monetization routes and the general fragmentation of the landscape with market-models and use-cases which posit telecoms companies, network providers, mobile operators or even dedicated commercial 'smart space' operators as the most likely route to market, highlights the 'bootstrap' problem inherent in the domain. Until there is some clarity in terms of how a market in UbiComp applications might function, few companies will be motivated to commit resources to producing such applications.

In such a situation, open source, user-community based development models are an attractive option. As an example of how such models can inspire further innovations in the commercial field, we only need to look at the Web 2.0 eco-system. One of the most important innovations in the domain is the focus on user-generated content, provided by engaged user-communities. These ideas were pioneered and developed by non-commercial user-communities, such as Wikipedia.org, and Slashdot.org and only later were they incorporated into commercial models by companies such as Blogger.com and YouTube.com. Due to the commercial strength of telecoms companies, and their traditional focus on 'locking-in' subscribers to their proprietary services and applications – the open community-development model has been slow to permeate in the mobile-device market. However, the rapid pace of development enabled by open approaches on the internet, accessible through Wi-Fi protocols that are increasingly supported by mobile phones, threatens the control that mobile operators have traditionally enjoyed over their subscribers. This pressure has prompted telecom operators and device manufacturers to increasingly embrace open community-based development models. For example, Apple's i-Phone is designed to allow the easy integration of third party applications. A step beyond this is the Android project of the Open Handset Alliance – a group of more

than 30 technology and mobile companies collaborating to produce a completely open and free mobile platform.

Thus, it is reasonable to say that many of the core technical problems which have obstructed the proliferation of UbiComp services in the real world are on their way to being solved. Cheaper sensors, more interconnected devices, better standards, wider support for common protocols and programming platforms, less lock-in to proprietary networks and better access to open information sources are all helping to make innovative and exciting UbiComp applications more viable. Many of the enabling innovations which unleashed the creativity of web 2.0 are becoming a reality for UbiComp applications. However, while this might address many of the factors hindering application creativity, it doesn't address the management problem, an area where we can't just look at the web for answers, since some of the problems are quite specific to the UbiComp domain.

A core requirement that is necessary to make many potential UbiComp applications viable is that the resources – sensors, networks, data sources and actuators – must essentially disappear from a management point of view. Many envisaged location-aware smart applications add very little value by themselves and will not be deployed if they require significant administration and management. Sensor and actuator networks are largely formed of cheap inexpensive components and unless the complexity of managing these networks grows less than linearly with the size of the network, the management will quickly become a bottleneck. For this reason, from a management point of view, the emphasis must be on autonomic, self-organising solutions.

However, while a system administrator would like the underlying resources to become essentially invisible, from a user's point of view, the management of UbiComp services must be visible and must allow explicit management in a number of ways. UbiComp applications which involve the use of sensor data cause privacy concerns amongst users which represent a significant barrier to acceptance of the technology [5]. When explaining the ideas of sensor-rich, ambient computing environments to ordinary users, it is common for them to invoke the concept of 'big brother' without prompting. Non-technical people, in particular, express anxiety when they find themselves in situations where they feel that their behaviour is being monitored and analysed by technological systems which they do not understand. This negative psychological reaction to applications and environments which use embedded sensor data represents a problem which needs to be addressed before UbiComp environments can fulfil their potential – they must be perceived as helpful rather than threatening.

User concerns about systems which monitor them are not merely irrational psychological reactions to technology. There are significant privacy and security risks associated with the sort of data that UbiComp applications depend upon. While a sensor network tracking an individual's movements might allow application designers to provide a range of helpful location-aware features to the individual, most people would not be happy to make detailed information about their precise locations publicly available. Such information is generally considered private and in certain situations it might constitute a grave security risk to allow the wrong person to access it – for example when an individual has a 'stalker' or when they are operating in an unstable security environment where politically or economically-motivated kidnapping is a real risk. There are also many less dramatic examples in which individuals have genuine privacy concerns about how their personal data will be used by governments and private corporations.

Beyond location monitoring, many other postulated UbiComp applications bring similar concerns with them. For example, health-monitoring systems which use simple sensors to measure and report on critical physiological indicators, such as heart-rate, blood pressure, body temperature and blood sugar levels have the potential to be extremely useful, but also pose a serious privacy problem. Most modern societies consider each individual's health-information to be private to that individual – indeed the concept of doctor-patient confidentiality is deeply embedded in the codes of practice which guide medical professionals. Therefore, it is obvious that the actual deployment of such systems will require that a satisfactory privacy management system be in place.

In general, any UbiComp application which collects data about people that was not previously available, creates an entirely new management problem. Who should be able to access this data? Since each application may be dealing with an entirely different human domain, involving different groups, each with quite different access control requirements, how can we properly control access control rights across the various applications which make use of the sensor data? How can we do so in such a way so as to fulfil the auditing requirements of UbiComp environment operators? How can we allow users of such environments to overcome their anxiety about being monitored and to come to trust these systems?

Finally, it is important to realise that UbiComp applications may frequently utilise sensors and actuators which are embedded within a physical environment and that physical environments are often shared spaces. Therefore, in delivering services to users, we must take into account the possible presence of other users within the same physical space. In certain applications, for example, the management of environmental controls (heat, light, etc), where there are multiple different preferences amongst users of the space, the application will need to, in effect, allow the users to arrive at a negotiated compromise.

3. SOLUTION: DECENTRALISED, PROGRESSIVE SELF-MANAGEMENT

In order to address the above problems, we adopt the following high-level approach. The best way for a service to gain users' trust, is to give them the perception that they are in control of the personal data which the service makes available and can, broadly, decide what happens with that data. The best way to give users this perception is to actually give them control.

This approach goes against the predominant approach to the management of personal information in the web-ecology. Typically, when a user signs up to a web-based service, they agree to allow their personal information to be used by the service-provider without any restrictions beyond those in national data-protection legislation. Many popular web-based services allow users to exercise some degree of access control over their personal data, but this is typically very coarse grained and inflexible. For example, the popular Facebook social network allows users to specify that various pieces of information relating to them should be publicly accessible or should be only visible to their "friends".

This essentially represents a binary division of the world's population into 'friends' and 'not friends' and a binary division of each user's data into 'public' and 'restricted to friends'. Such coarse granularity and inflexible grouping is incapable of dealing with the access control requirements of personalised health-data derived from sensors, not to mention the problems that might arise if a UbiComp service provider claimed ownership of such intimately personal data. Even on the web, although providers typically claim rights to use the data they collect however they please, in practice, they rarely do this. Services which sell their users' personal information to marketing companies are generally not looked kindly upon by the public and it is becoming increasingly common for providers to forsake such rights in order to build trust amongst their users. Even in situations where a user's personal information is already available, collating and aggregating it and making it public can be enough to arouse user-resistance. For example, the launch of Facebook's *Beacon* advertising system caused a storm of protest amongst users, which eventually caused the company to turn *Beacon* off by default [10].

In contrast to the standard, centralised solution to application management, we propose a decentralised management system, where users progressively define how their personal data is available to others and to third party applications through negotiation between users and application providers. We envision that democratised, participative management will help overcome user-concerns about data privacy and loss of control. However, there are a number of significant challenges in delivering a system which can empower users in such a way that its impact upon their acceptance of the technology can be properly evaluated.

- Participation in management tasks must not be onerous or difficult to learn. Users need engaging tools to allow them to assess the impact of their management decisions.
- Management tools must strongly support collective decision making since physical UbiComp environments have diverse user and stakeholder communities. Management decisions cannot therefore be taken in isolation by each user.
- Participation must yield visible progress to users. This means that their concerns must be addressed in a measurable way so that the results of their participation become clear to them.

In order to evaluate the viability of our approach we are applying it to a number of case-studies that demonstrate the real-world potential and challenges in using sensors to collect information and to give control over that data to the subjects of the data collection. These case studies will allow citizens to decide how to use sensor data collected in their own homes. This scenario addresses the most pressing human level challenges facing ubiquitous computing: the need to provide transparency and control at the level of the individual user and hence generate sufficient trust to allow the technologies to be widely deployed. The aim is to develop and apply hardware, software and user interfaces that are easy to use and make transparent to and give control to the citizen about what information is collected, transferred and to whom. We will tailor and apply Information and Communication Technologies (ICTs), specifically those developed for Digital Homes (DHs) and other Ambient Intelligence (AmI) technologies to allow individuals living in a household to easily and transparently collect data on the households and individuals within that household's behaviour. Such domestic sensor data is useful for providing empirical evidence for a variety of public policy issues, e.g. energy consumption, family activity levels, and also for providing direct support for citizens, e.g. security or fire monitoring and home health monitoring. In our case study, we are focusing on energy monitoring, specifically monitoring the power consumption of key appliances such as heating/cooling, lighting, hot water, washing machines, TVs.

Using the technology of digital homes, in which networked devices coordinate actions, higher order functionality can be built, making so called smart or intelligent homes [2]. Apart from enabling coordination, the networked aspect of the appliances allows sensing or monitoring of the usage of appliances or services. This aspect of digital homes makes them amenable to providing accurate and timely data on the usage of home appliance (e.g. the energy usage behaviour of a family such as what power is being consumed, by who, by what, where and for how long). This feature of digital homes allows them to act as a source of usage and behaviour to inform stakeholder (e.g. energy companies) and government policy formation processes. In this respect a helpful analogy is to consider homes as micro models of government where small communities, a family, are governed by people (e.g. parents) to effect agreed policies (e.g. acceptable behaviours) [1]. The next step would be for the family or household to engage in deliberation with other households to produce agreed policies for managing the data, i.e. who to share it with and for what reasons. In effect, the proposed system will explore how micro policies within the home and neighbourhood (geographical or virtual) can inform stakeholder and government policies at a national macro level (and vice-versa). In effect, it allows the citizen to engage in an empowered and autonomous fashion with fellow citizens, relevant stakeholders and policymakers.

The case study hopes to demonstrate that the best way to protect and empower the user is to make the operation of the system transparent and maximising the users ability to make choices on what information is gathered and when and how it is used (including choices to 'anonymize' the collected data). This requires and enables groups at various levels of granularity (from families to neighbourhood, town and even to nations) to e-debate/deliberate what data should be made available to whom for what purposes. Cheap storage, distributed systems querying and perhaps p2p based backups could mean that sensor data could be stored relatively locally (minimising exposure to massive theft and allowing the levels of security and robustness to be tailored to the group concerned).

Debate and deliberation over the use of the data and the policy issues concerned is a central part of the model proposed. A critical challenge is the feasibility of the model of deliberation as it is scaled up from household to local to municipal to national to regional to inter-national levels. Virtual or online deliberation, although not necessarily the only type of deliberation involved, is a crucial factor. The challenge is to make it feasible for deliberation on the use of this data to be scaleable while ensuring the data will be owned and managed collectively by the communities about whom it is concerned. This will enable them to debate the level of exposure, and tailor specific policies to individual and sub-group concerns.

Deliberation online – or e-deliberation – is seen as having the potential to reinvigorate the public sphere. It certainly provides a means for broadening the concept of citizen involvement in

policymaking, and protecting democracy and citizenship autonomy in a knowledge-based society. According to Roza Tsagarousianou, "new technologies clearly have the potential to sustain such spaces [public spheres] as they enable both deliberation (citizen to citizen communication) and "hearing" (citizen to authority communication)" [12]. The Internet, especially through asynchronous discussion forums, can offer easy solutions to four traditional problems which have prevented people from fully participating in public debates because of the constraints they impose: time, size, knowledge and access [11].

However, there are also a number of factors limiting the expansion of the public sphere online. They include the increasing colonisation of cyberspace by state and corporate interests, a lack of reflexivity, a lack of respectful listening to others, the exclusion of many people from online political forums, and the domination of certain individuals or groups [3, 6]. In addition, the software and the type of interaction it allows, the nature of the moderation, the appropriation of the interface by users and their ability to manipulate it: all of these elements have been shown to intervene in the success or failure of a deliberative experiment. The presence of moderators affects both how citizens participate [14] and how their participation will impact the final decision [4, 14].

While there are threats to privacy, there are also benefits, both of which need to be factored in to the decision making process of citizens. For example by participating in such energy usage monitoring programmes ordinary citizens can both help the environment and save money by using less energy. In addition, as such information might help make the energy providers more efficient, these savings could be shared with the consumer in the form of a discount providing additional incentives for the citizen to participate. It is also important to emphasise that UbiComp applications may have many stakeholders – individuals, voluntary organisations, governments and corporations, for example. Our case study has the potential to lead the way in enabling user-control in critical areas, such as the citizen's relationship to the state. However, in order to make such control viable, we need to augment e-deliberation tools with user-centric relationship management capabilities. The following section describes the platform which we have implemented to provide such capabilities.

4. CBPMS-PUDECAS PLATFORM FOR PROGRESSIVE SELF-MANAGEMENT OF UBICOMP ENVIRONMENTS

We have constructed a software platform which allows users to manage their own participation in the various services provided in the space, in collaboration with the other users of the space. This platform is based on the integration of PUDECAS, a UbiComp simulator, with the Community-Based Policy Management System (CBPMS), a distributed policy management framework. A messaging tool has also been integrated into the platform. Pudecas allows service designers to create rapid realistic simulations of UbiComp environments, while the CBPMS allows users to manage the services in that space collectively. This platform allows application designers to evaluate how user-acceptance is affected by the degree of self-management.

4.1 Community Based Policy Management

The CBPMS provides a framework with which to address the key challenges in managing online deliberation. It is based on a novel mechanism developed to integrate resource management with collective decision making. CBPMS differs from previous policy-based management approaches in its novel use of self-defining groups as the fundamental structural abstraction as opposed to centrally defined roles typical of access control systems. A group is simply established by a set of people engaged in a shared activity. By defining sub-groups and federated groups through explicit mandates for exercising decision-making authority, an organisation of self-managing groups can be formed around the evolving needs and experiences of a user-community. These mandates can be progressively grounded as patterns of authority change or as new models of resources or context emerge and their impact on the distribution of authority is learned. Controlling the interconnection of different social and IT management rules between groups restricts the portion of an organisation's current rule set that collaborating decision-makers must understand, thereby making collective decision-making more scaleable. This also delivers fast runtime policy rule checking. Clear providence of rules ensures that the causes of conflicts when policies authored in one part of a community clash with policies or goals from another are immediately identified, thereby quickening their resolution. The explicit modelling of group structure and policy-rules and the resulting identification of policy conflicts and the parties necessarily involved in their resolution, therefore enable communities to actively reflect on their management processes.

The CBPMS is service-oriented system that can be easily integrated with other resource management and e-deliberation solutions. It provides a means through which users can form and federate groups that can then electronically manage, reflect and debate the best use of ambient intelligence resources in servicing their own needs and that of the wider community. By rendering such activities explicit, while also providing user friendly web tools to browse and manipulate the related models, this approach enables communities to decide clearly how authority to collect, monitor and process sensor data should be distributed.'

Significantly, this approach means local groups can negotiate about access of their data by 'outside' bodies, e.g. governments and corporation, from a position of power. These bodies would therefore be forced to argue for access to data on the basis of earned trust, transparent procedures, and well reasoned appeals to the common good or appropriate incentives. Agreement for access could be provisional (these are on-going feeds of data so long-term relationships are key) and linked to systems for auditing security and usage and also for dealing with potential conflicts that may arise, e.g. if one accessing bodies wishes to forward data to another one (a big concern with a lot of data privacy - you may trust the local police but not want them to pass details to the CIA).

4.2 PUDECAS UbiComp Simulator

PUDECAS is a 3D simulation environment designed to test context-aware adaptive services in the wireless, mobile and context aware markets. It was chosen for this experiment as it provides a realistic environment for testing ubiquitous computing systems without the expense of fitting an entire building with sensors and activators. It provides service developers with a toolkit for creating 3D simulations of physical spaces which allow events within the simulator to trigger calls to external, networked services. It is based upon the well-known Valve Source Engine from the popular Half-Life 2 computer game. The PUDECAS application framework has been introduced in detail in [9].

Figure 1 Screenshot from PUDECAS Simulation of Trinity's Lloyd Building and Actual Photo

4.3 Messaging Tool

PUDECAS is primarily designed to provide a visual experience of physically travelling through a UbiComp environment. Due to the fact that its underlying engine is geared towards game playing, it has limited support for modelling sophisticated personal messaging devices, such as mobile phones and PDAs. Therefore, a general purpose Instant Messaging (IM) application was also integrated into the system. This IM application communicates via a JABBER IM server, using the Extensible Messaging and Presence Protocol (XMPP) [7]. The IM client supports the simulation of personal messaging devices, and allows users to send messages between each other and to access an extensible range of information services, such as location tracking.

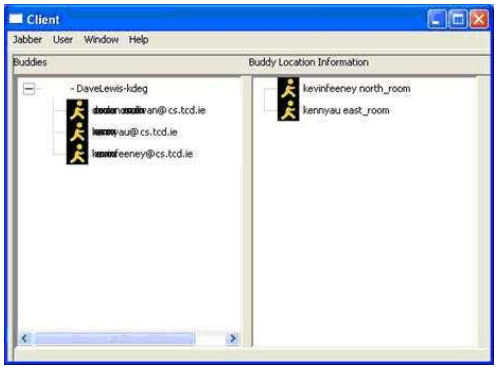

Figure 2. Screenshot of IM Application showing location-tracking. The left pane shows the current users of the system, the right pane shows those whose locations are available

4.4 CBPMS / PUDECAS / IM Integration

Figure 3 below shows the network architecture of the integrated UbiComp service evaluation platform.

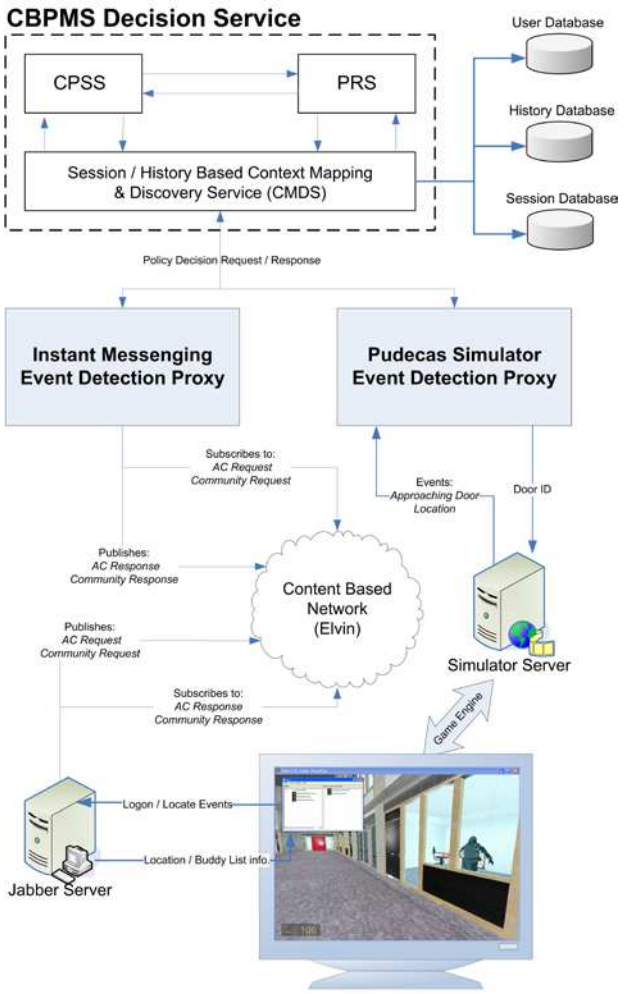

Figure 3. Network architecture of the UbiComp platform. As the User moves through the space, doors open and locations become visible according to policy defined in the CBPMS. s

The IM JABBER platform publishes events to a Content Based Networking (CBN) infrastructure based on Elvin servers, through an IM interlocutor – an application which intercepts IM client requests and can invoke external services, including JABBER. This architecture allows the routing of messages between collaborating services to be undertaken by a decentralized network of content-based routers rather than relying on a single IM server due to the benefits that this arrangement can provide in terms of robustness and load-sharing. CBNs provide content-delivery via a publisher/subscriber model. The IM application allows each user to request presence information of all the other users of the simulator – in particular location information – and to communicate with each other (all depending on the policies that have been agreed on).

The CBPMS uses two event detection proxies to communicate with the IM application and with the PUDECAS simulator. In the case of the IM application, the CBPMS proxy simply subscribes to the relevant messages on the CBN and publishes policy

decisions, labelled accordingly, to the CBN, which are then read and carried out by the application. In the case of the PUDECAS simulator, all relevant events within the UbiComp environment are published to the CBPMS event detection proxy, and these are mapped to policy decision requests. In situations where the decision must be transmitted back to the simulator (for example, when a policy decision dictates that a door should be opened), the proxy sends an appropriate message back to the simulator, which then carries it out (for example, by opening the relevant door).

This architecture enables the services offered by the UbiComp environment and by the personal messaging device to be managed by the CBPMS through the collaborative authoring of policy rules. The CBPMS transmits policy decisions to both the simulator server and the Jabber server. By following these policy decisions, the simulator becomes managed by the policy system. The rules defined by users, which are arrived at through a process of negotiation on the CBPMS, thus have an immediate and direct affect on the behaviour of the simulated environment. The policy rules defined in the CBPMS can leverage information culled from external information services, such as user-databases, session databases or any other information that is relevant and available.

5. EXPERIMENTAL SCENARIO

We have carried out several experiments to test the UbiComp management evaluation platform, based on the architecture described in the previous section. These experiments were based around similar scenarios involving the collaborative management of services within a simulated UbiComp environment.

The scenario involves the management of a building used as a research institute within a University, housing teams of scientists working on different projects, some of which are of a confidential nature. The institute houses their research laboratories, their personal offices and common areas such as the lobby, stairways and canteen. The building is a UbiComp environment, where the location of each person is tracked by sensors. Doors automatically open for those who have appropriate access rights and each user's location is available on an instant messaging. The following were the basic requirements:

- There is a variety of legislation and various rules of the University which govern physical access to buildings and this must be guaranteed by the system. For example, fire doors must allow access in emergencies.
- All of the groups working within the building consult in the formation of policy governing common areas.
- Each research group decides who is allowed access to their laboratories.
- Individual researchers decide who is allowed to access their personal offices, except in exceptional circumstances.
- Individual researchers will be allowed to control access to their personal location information, except in exceptional situations, as decided by University rules.

For the purpose of this experiment, a simulation of the Lloyd Institute in Trinity College Dublin was constructed, as shown above in Figure 1. This happens to be the building in which the simulator was developed and it provides a sufficiently large and detailed model of a realistic environment to test a wide range of UbiComp services.

5.1 Creating the CBPMS Models

Our experiment requires that access to all of the doors, the population of people's buddy lists, and access to their location information is to be managed by the CBPMS. The first task was thus to create models of these resources suitable for integration into the CBPMS resource management system. This is a straightforward task as the managed resources are extremely simple. The doors in the building can be modelled as a simple hierarchical tree of doors, gathered together into convenient grouping nodes. The grouping nodes in the tree allow policies to be specified for groups of doors more easily, for example, by room and by floor.

The only other resource that is to be managed by the CBPMS is the location information of each user. Once again, this is very simple as there is only a single action that users can take vis-à-vis another user – to view their location. Rather than building a static representation of the users in the system, however, the resource model merely provided an interface into the user database utilised by the IM interlocutor. This option was taken in order to allow the system to be easier to maintain, since users only had to be registered in a single, system-wide database from which the resource model was dynamically constructed. However, in order to provide support for new ubiquitous services and future modifications to the simulator software, the action trees of both resource models were extended to include various actions which, while not being supported by the current software, might be supported by future ubiquitous services using the simulator. Thus the resource models constructed, pictured in Figure 4, extended the action trees to include various extra nodes.

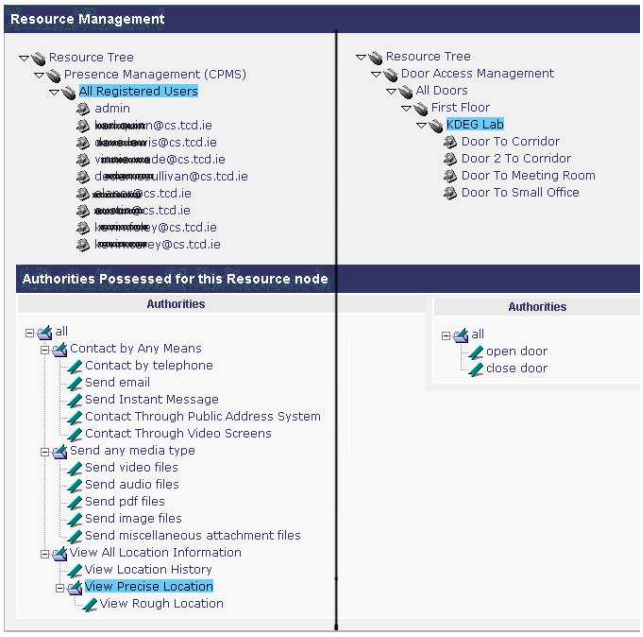

Figure 4. Resource Models (montage from CBPMS GUI). Left panel shows the resource tree for the doors in the simulation; right panel shows the resource tree for locations.

Having constructed the resource models, the next step is to design a basic community, based on the organisational structure of the building's user-community, as derived from the scenario requirements above. The model is pictured in Figure 5 and is loosely based upon a cross-section of those groups within the

department of computer science in Trinity College Dublin who currently use the Lloyd Institute building. It was created by recursively breaking down the users of the building into more specific and smaller units. It is not intended to form a definitive static, organisational specification, rather it can be seen as a basic structural starting point from which the organisation can evolve, by forming new communities and hierarchies of communities.

Figure 5. Segment of UbiComp Experiment Community Model

5.2 Self-Management through Delegation

Having created the initial basic model of the communities, in order to enable self-management, a sequence of CBPMS primitives is invoked in order to delegate appropriate authority to the users, to seed the system. Each resource managed by the CBPMS has an owner within the CBPMS model. Resource owners grant groups and users access to their resources by delegating *resource authority* to the appropriate community. Resource authorities represent authority to carry out a well-defined subset of the capabilities that are supported by the resource. They are composed of a node from each of the trees that make up a resource's hierarchical model. For example, the pair [*all registered users, view all location information*] is a resource authority which grants authority to access all location information about all users of the system.

In this particular example, we can model complete self-management of each user's presence information by assigning ownership of the *[username, all]* resource authority to each user themselves. Since the doors are ultimately a collective resource, we assign ownership of them to the root of our community hierarchy. Once the ownership of the resources has been allocated, access to them, and management control over them, can be shared by means of delegation. Thus, each user can delegate access to a subset of their presence information to any community in the hierarchy and the entire group can delegate management responsibility over certain doors to particular communities. For example, we might give each research group authority over the doors to their own labs. CBPMS delegation allows access and management rights to be shared across a user-community in a controlled way. The control comes from the fact that specific subsets of the capabilities of the resource are delegated – meaning that other capabilities are excluded – and from the fact that the owner of a resource can apply policy rules to a resource authority, constraining its usage by those who have been delegated it. Groups and users who have been delegated resource authorities can themselves add policy rules and further delegate a subset of the authority which they have been granted.

In the case of this experiment, having delegated the appropriate authority to the various groups, policies were put in place to enforce the management requirements. The management community specified policies in the root community which implemented the various legal codes and building regulations governing access. Thus, for example, a policy was defined for all of the access doors to the building specifying that access should be permitted during emergencies. The *staff* community defined a policy which specified that all office-holders could specify their own policies about their personal office doors (this had to be specified in the *staff* community so that it would be scoped to apply to all staff). The various research groups specified policies which should apply to the doors that were delegated to them. The *management* community defined a policy specifying that the location information of all the building users would be available to management when they were in the public areas of the building (the lobby, the canteen). The *academics* community defined a policy which specified that each member had permission to specify policies about who could access their personal presence information. All of these policies were specified by attaching a resource authority to a simple, unconditional 'permit' or 'deny' policy.

One element of the CBPMS model that is important to appreciate is the fact that the communities in the model are themselves considered to be managed resources just like any other resource. Therefore, each community can be given authority to manage its own structure, form sub-communities, define decision making rules, in a dynamic way as requirements are identified and later change. The combination of ownership, resource authorities, delegation and community self-management enables a progressive and participative approach to management of the services. Application designers can limit the degree of self-management, by only assigning certain capabilities to self-managed groups, or by authoring policy rules which constrain how self-managed resources can be accessed.

5.3 Experimental Evaluations

We have carried out several experiments using the architecture and scenario described above. Each experiment involved 4 users, none of whom were familiar with the underlying technologies, navigating their way through the simulated environment. Each user was assigned to a set of communities in the model, by adding them to the simple membership rules of the appropriate communities and these community memberships were allocated so that each user shared membership of a research project with another user. In each iteration of the experiment users were assigned various tasks, such as attempting to gain access to a room or a corridor in the building, or attempting to track each other's locations as they moved through the environment.

The first run of the experiment was used to familiarise the experimental subjects with the system, and to fine-tune the policy set and the software implementation. The success or failure (e.g. because they could not enter a door) of the users in their tasks in each of their allocated tasks was observed and compared to the results expected. Metrics were also gathered from the experiment, as well as from subsequent experiments, which recorded the number of policy rules that had to be evaluated for each policy decision request and the time that it took to return policy decisions. The second run of the experiment involved testing the modifications that had been made in the previous run and introducing new tasks to check for any undiscovered specification problems. The third run of the experiment involved modifying the community set to purposely introduce several conflicting policies into the system and setting users tasks designed to trigger

these conflicts in order to test the CBPMS's ability to automatically resolve conflicting policies as well as to highlight the organisational source of irresolvable conflicts. The fourth and final run of the experiment involved testing the system's ability to respond in real time to modifications in the policy set and community models. As the users navigated through the environment, policy rules were modified in real time, sub-communities were created and destroyed, and membership rules were changed and the observed behaviours were compared with the expected behaviours.

The users found it easy to familiarise themselves with the simulated environment due to the fact that they were already familiar with the general layout of the building and the user interface of both the IM application and the PUDECAS simulator were easily mastered. Their general familiarity with applications that use the Source Engine (such as the popular computer game "Half-Life") undoubtedly aided matters.

The policy set also proved generally robust; however, some small problems were identified when doors did not open as they were expected to. 3 policy rules had to be removed and 15 policy rules added to the policy set in order to correct these specification problems. The problems generally involved forgetting to specify rules for certain doors, or specifying a rule in a community that was too far up the tree which caused conflicting policies, defined further down the tree to be ignored. For example, in one case, a deny policy was specified for a door, but a permit policy applied to the entire floor in a parent community, meaning that the deny policy was never invoked – a problem solved by moving the deny policy up one level in the community hierarchy Several of the tasks set for users were designed to validate the fact that the CBPMS system served as an accurate implementation of the CBPM decision algorithm, in particular that it would react correctly when it encountered conflicts.

The final run of the experiment involved an administrator updating the policies of the communities, changing the community structure and changing the membership rules of communities. These changes were tested by asking users to attempt to access doors, making changes and then asking them to attempt to access the doors again and observing whether the behaviour of the doors changed to reflect the changes to the model. All of these changes were successful. The users were able to access doors according to the organisational model that existed in the instant that they approached the door and triggered the event. The opening of doors reflected the dynamic model of the organisation and how the user fitted into it. In no cases was it possible to identify any discrepancies between the organisational model as it changed and the expected behaviour observed, from careful analysis of the server logs.

Although these experiments provided some evidence of the ability of the CBPMS to model organisations with a dynamic and flexible structure, it also revealed several shortcomings in the user interface with respect to such a dynamic environment. The relative complexity of the policy authoring user interface, in particular the composition of multiple rules into a single XACML policy, meant that several 'clicks' were required for each modification. The fact that the interface was web-based and was running on the same server as the rest of the CBPMS services caused each page to take up to 5 seconds to load. Since most changes involved several different actions (e.g. creating a new community, specifying a membership rule, specifying a policy, specifying policy rules, specifying a condition), each modification took up to a minute to carry out, and while this is quick in most domains, in the world of simulators based upon computer game engines, whose users are accustomed to a highly dynamic experience, one minute waiting for a change to be made before carrying out a task is an unacceptable length of time, evidenced by the restlessness of the users. Therefore, while the experiments showed that the CBPMS was capable of modelling rapidly changing dynamic organisational structures in real time, the web-based management interface prove insufficient for real time policy authoring in such a dynamic environment.

Due to the relative complexity of the user-interface, and the lack of familiarity amongst the experimental subjects in policy authoring, it was not possible to gain a detailed evaluation of the direct affects of particular aspects of self-management on user-confidence. Because of the complexity of the management system, which allowed users to author policy rules in fully-featured policy languages most users found authoring their own policies to be relatively difficult. Nevertheless, users of the system unanimously agreed that the fact that they had the ability (in theory at least) to exert fine-grained control over the use of their personal data and the fact that they had the ability to test their participation in a simulator would increase their willingness to participate in similar UbiComp services if they were rolled out in the real world.

6. CONCLUSIONS & FUTURE WORK

In this paper we have described a general-purpose high-level approach to the problem of overcoming privacy-related fears amongst users, with the goal of increasing user-participation in UbiComp applications. We have shown the operation of our simulation management platform, which allows users to explore a UbiComp environment and evaluate the impact of different management policies in a realistic simulation of a UbiComp environment. It is our intention to continue to develop our platform in order to provide a convenient toolset which will allow UbiComp service designers to better understand their users privacy concerns and where the best trade-off between self-management and centralised management lie. However, there are a number of remaining challenges which must be addressed.

Firstly, we need to integrate the CBPMS into a broader e-deliberation framework. Due to the fact that it derives from technology developed within the network management community, its management system is not well suited to relatively non-technical users. We are working on a range of tools which will provide intuitive interfaces to users which will allow them to get a better grasp of the semantic significance of the various policy options available to them. These tools include policy-visualisation, online negotiation, a variety of communication tools and a 'docket' system, which allows users to request changes to group decisions, in a structured way.

Secondly, we are extending PUDECAS to handle more sophisticated tools for placing and configuring sensor systems within Virtual Worlds and presenting these to potential stake-holders. PUDECAS offers a UML based tool built on the eclipse GMF (Graphical Modeling Framework) to place and configure simulated sensor systems in a Virtual Environment.

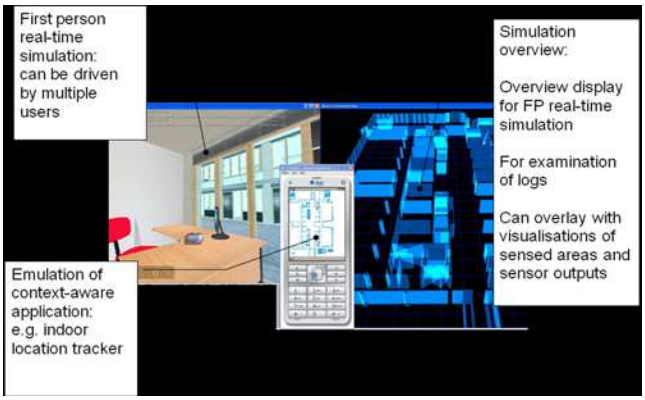

Fig 6 Left: Virtual Environment, Centre: a J2ME Emulator for Indoor Location Tracker, Right: Visualisation Tool

These simulations provide a means by which developers of smart spaces can present sensor data collection non expert users in a meaningful and open manner. To improve the process of presentation and make it accessible to non expert users, a visualisation tool has also been designed and a prototype implemented (Fig. 6). The tool presents a simulation overview of the smart space (e.g. a building environment) and presents real time location updates visually (the location of the user in the VR Environment) as well as sensed location updates (the location as sensed by a simulated sensor system).

To improve this process we are integrating the tool with standard data models used architectural and facilities management tools. A promising standard is the IAI (International Alliance of Interoperability) *Building Smart Mission* which has developed the IFC (Industry Foundation Class) ifcXML Common Model. This is a neutral data format to describe, exchange and share information typically used within the architectural and facility management industry sector.

We are also looking at the use of Collado (COLLAborative Design Activity), an XML based open standard for the exchange of digital assets among various graphics software, to model the 3D Virtual Environments. The first person VR Environment in conjunction with the Visualisation Tool offer a powerful and visually captivating method for presenting Smart Spaces to a wide range of users, who are then in a position to critically analyse the impact of location sensing technologies in their home or business environments. Additional management tools can potentially then further empower users by allowing them to choose the types of contextual information on display, when and to whom.

Finally, as we further develop these tools, we need to ensure that where users need to be involved in management, there needs to be emphasis on interfaces for ordinary people. In our work in NEMBES, in particular, we will examine how semantic mapping and policy refinement techniques can be used to enable the expression of high-level user-centric goals using simple form web-forms and automatically translate these into low-level technical policies and vice versa. We will perform a range of increasingly sophisticated experiments designed to analyse the relationship between degrees of self-management, user-perceptions of control and user-participation in UbiComp applications. This research will provide a wealth of valuable data for service designers wishing to utilise data derived from sensors without coming across as 'big brother'

7. ACKNOWLEDGMENTS

This work was partially supported by Enterprise Ireland through the PUDECAS project and the Irish Higher Education authority under the PRTLI4 project NEMBES.

8. REFERENCES

[1] Callaghan V, Clarke G, Chin J. 2007. Some Socio-Technical Aspects Of Intelligent Buildings and Pervasive Computing Research, Intelligent Buildings International Journal, Vol 1.1

[2] Callaghan V, Clark G, Colley M, Hagras H Chin JSY, Doctor F. 2004. Intelligent Inhabited Environments, BT Technology Journal , Vol.22, No.3 . Kluwer Academic Publishers, Dordrecht, Netherlands

[3] Dahlberg, L. 2001. Computer-Mediated Communication and The Public Sphere : A Critical Analysis. Journal of Computer-Mediated Communication, 7.

[4] Edwards, A. 2002. The moderator as an emerging democratic intermediary: The role of the moderator in Internet discussions about public issues. Information Polity 7, 3-20.

[5] Iachello, G., et al. 2005. Control, deception, and communication: Evaluating the deployment of a location enhanced messaging service. Ubicomp 2005.

[6] Jankowski, N. W. & R. Van Os 2004, Internet-based Political Discourse: A Case Study of Electronic Democracy in Hoogeveen", in P. M. Shane (ed). Democracy Online. The Prospects for Political Renewal through the Internet, pp. 181-193, New York: Routledge.

[7] Kenny, A., Lewis, D., O'Sullivan, D. 2006. Interlocutor: Decentralised Infrastructure for Adaptive Interaction, 3rd International Workshop on Managing Ubiquitous Communications & Services (MUCS 06), Cork, Ireland.

[8] O'Neill, E. et al, 2005. A Testbed for Evaluating Human Interaction with Ubiquitous Computing Environments. Proc. IEEE 1st International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities.

[9] Quinn K. et al. 2006. A Framework for the Decentralisation and Management of Collaborative Applications in Ubiquitous Computing Environments, 10th IEEE/IFIP Network Operations and Management Symposium, Springer.

[10] Story, L. and Stone, B. 2007 "Facebook Retreats on Online Tracking", New York Times, November 30, 2007

[11] Street, J. 1997. Remote Control? Politics, Technology and 'Electronic Democracy', Journal of Communication, vol. 12(1), 27-42.

[12] Tsagarousianou, R. 1999, "Electronic Democracy: rhetoric and reality", Communications. The European Journal of Communication Research, 24 (2), 189-208.

[13] Wojcik, S. 2007. Les modérateurs des forums de discussion municipaux, des intermédiaires démocratiques? Questions de communication, n° 12, Presses Universitaires de Nancy.

[14] Wright, S. (2005). Design matters; the political efficacy of government-run discussion forums, in R. Gibson, S. Oates & D. Owen (Eds), Civil society, democracy and the Internet: a comparative perspective, pp. 80-99, London: Routledge.