

# Exploring the Design of Pay-Per-Use Objects in the Construction Domain

Daniel Fitton, Vasughi Sundramoorthy, Gerd Kortuem, James Brown, Christos Efstratiou, Joe Finney and Nigel Davies

Computing Department, InfoLab21, South Drive, Lancaster University, Lancaster, UK  
{df, v.sundramoorthy, kortuem, jb, efstrati, joe, nigel} @comp.lancs.ac.uk

**Abstract.** Equipment used in the construction domain is often hired in order to reduce cost and maintenance overhead. The cost of hire is dependent on the time period involved and does not take into account the actual use equipment has received. This paper presents our initial investigation into how physical objects augmented with sensing and communication technologies can measure use in order to enable new pay-per-use payment models for equipment hire. We also explore user interaction with pay-per-use objects via mobile devices. The user interactions that take place within our prototype scenario range from simple information access to transactions involving multiple users. This paper presents the design, implementation and evaluation of a prototype pay-per-use system motivated by a real world equipment hire scenario. We also provide insights into the various challenges introduced by supporting a pay-per-use model, including data storage and data security in addition to user interaction issues.

**Keywords:** Smart Objects, Pay-Per-Use, User Interaction, Mobile HCI.

## 1 Introduction

The application of pay-per-use payment models is receiving growing interest and is perhaps exemplified by numerous existing road pricing systems deployed around the world and various examples of pay-per-use car insurance [1][2]. Pay-per-use or ‘metered service’ is a payment model that allows customers unlimited access to services (such as electricity, phone calls, internet access etc.) or physical objects where cost is calculated according to usage. Many new examples of pay-per-use systems in use today have been enabled by advances in technology and motivated by economic factors (in the case of car insurance) or social factors (in the case of congestion related road pricing). While many examples of ‘metered service’ already exist in our daily lives the application of a pay-per-use model in the broader context of everyday objects has received very little attention. Pay-per-use can potentially be applied to any object into which a system to record usage can be embedded. To date, the only example of a pay-per-use object intended to investigate this idea is a prototype chair that contains sensors and can record use [3].

We explore this new research area by applying the pay-per-use concept to an equipment hire scenario in the construction domain. When construction companies deal with short-term contracts, it is common for equipment to be hired for the duration of that contract (in order to avoid purchase cost and maintenance overhead). However, the hire cost is currently proportional to the time period involved so regardless of whether equipment is used heavily or not at all the construction company will be charged the same amount. Additionally, those responsible for managing the hire of equipment have no accurate knowledge of how much use equipment is receiving and whether project management decisions need to be revised, for example, in order to reduce cost or increase productivity.

Our approach to addressing these problems is to augment individual pieces of equipment with sensing, storage and communication capability to enable them to become 'smart' objects. This work builds upon existing research using smart physical objects for safety monitoring on construction sites [4] [5] and we use smart objects to detect and store *experiences*. We use the term *experience* to refer to events or activities involving the object which it can detect. Detection of usage experiences in particular is then used to implement a pay-per-use billing model for equipment hire.

Smart physical objects not only allows the implementation of services such as pay-per-use but also provides the opportunity to enable a range of novel user interaction experiences. A piece of equipment used in the construction domain is involved in a range of meaningful experiences each connected with a variety of contextual information. Our goal is to make this information accessible to an interested (and authorised) party interacting via a mobile device. For example, a supervisor finding a discarded piece of equipment at a site could interact with it to find information such as whether it is in working order and how much its hire has cost to date.

Section 2 provides an overview then analysis of the construction scenario which motivates this work, followed by discussion of our pay-per-use billing model. Section 3 discusses the design of the pay-per-use smart object architecture. Section 4 provides an overview of the hardware and software implementation of the prototype system. Section 5 provides evaluation and discussion of the prototype including areas for future work. Section 6 discusses related work and section 7 presents concluding remarks.

## **2 Pay-per-Use for Construction Equipment**

### **2.1 Construction Scenario: Road Patching**

The design of our pay-per-use prototype is motivated by the scenario of road patching (where defects to tarmac road surface are repaired). The task of road patching begins with a contract between a governmental body such as a local council and a construction company. The local council then provides details of the roads that need to be repaired. Typically, once the contract for work is in place, the construction company (the lessee) hires necessary equipments from a hire company (the lessor) for the duration of the contract. Figure 1 shows a site where road patching is taking place and hired the equipment in use.



**Fig. 1.** Key Equipment Used for Road Patching: 1 – Van, 2 – Petrol Powered Drum Roller/Air Compressor (Stored on Trailer), 3 – Air-powered Pavement Breaker, 4 – Petrol Powered ‘Wacker’ Plate Compactor.

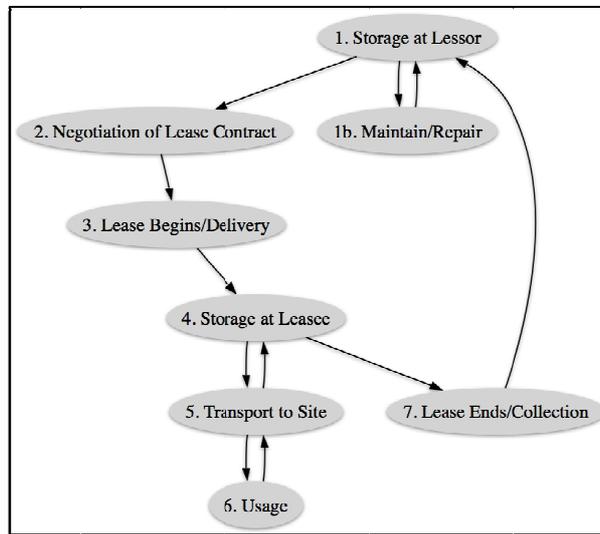
### 2.3 Analysis of Usage Scenario

Figure 2 depicts the main stages in the hire operation of a piece of equipment in a construction scenario. Within Figure 2 a range of users are involved, each with different responsibilities and goals. We divide users into two main categories. Firstly, employees from a company that lease equipment (stages 1, 1b, 2, 3 and 7). Secondly, we consider employees from a construction company that hires and utilises equipment with responsibilities summarised as:

- Administrator: Procuring and managing the hire of equipment (stages 2, 3 and 6),
- Supervisor: Managing and overseeing the deployment of workers and equipment (stages 4 and 5),
- Worker: Transporting equipment to a site and utilising it to carry out road patching (stages 5 and 6).

For the design of a smart object in a this scenario, three main areas to address can be identified; experiences the smart object must detect, user interactions that take place with the smart object and processes with which the object must take part. One example of each is discussed in the following paragraphs:

Usage Experience - The key aspect of the scenario for pay-per-use is the detection of usage experiences once workers have transported equipment to the required site. The equipment continually senses its state and stores usage experiences. If a piece of equipment develops a fault (which can not be sensed internally by the tool) a supervisor would interact to record this information. If an exception condition is sensed during usage (such as overheating, exceeding duty cycle, dropping/subjecting to excessive forces etc.) this mishandling experience is also recorded.



**Fig. 2.** Overview of Stages in Hire Operation of a Single Piece of Equipment

**Query Interaction** - This occurs when a supervisor visiting a remote site needs to view how much use a piece of equipment has received and the associated cost. The supervisor locates the equipment (which may be on a van or in the field) then interacts. In addition to cost, the supervisor is also able to view daily breakdowns of use (including per-worker) and any faults that have been reported.

**Delivery Process** - This concerns delivery of equipment by a hire company to a construction company depot after negotiation of a lease contract. At the point of delivery an employee of the hire company (lessor) interacts to confirm information relevant to the hire is stored on the device (date of delivery, cost information, point of contact at hire company etc.) and ensures that usage counters are reset. An administrator from the construction company (lessee) interacts to check equipment fulfils the required technical specification and confirm that delivery has taken place.

## 2.4 Pay-Per-Use Billing Model

A wide range of possibilities exist for calculating cost from usage. For example, charges can vary based on times of use (with peak and off-peak periods), usage level (such as cost reducing and usage increases) or even the intensity of use. We use a billing model which considers an object's usage and mishandling experiences. Equipment mishandling includes overheating, exceeding duty cycle, dropped/subjecting to excessive forces etc. We assume that if a piece of equipment is mishandled by the lessee additional maintenance (carried out by the lessor) is required and therefore a charge should be levied.

We now give the formal description of our own pay-per-use billing model. *LeaseCost* is the summation of the usage cost (where T is the total period of use and U is the equipment-specific cost per usage unit) and total costs for every type of mishandling experience (*M* is the set of mishandling experience that occur,  $m \in M$ ):

$$LeaseCost = (T \times U) + \sum_{i=1}^{|M|} Cost(m)_i \quad (1)$$

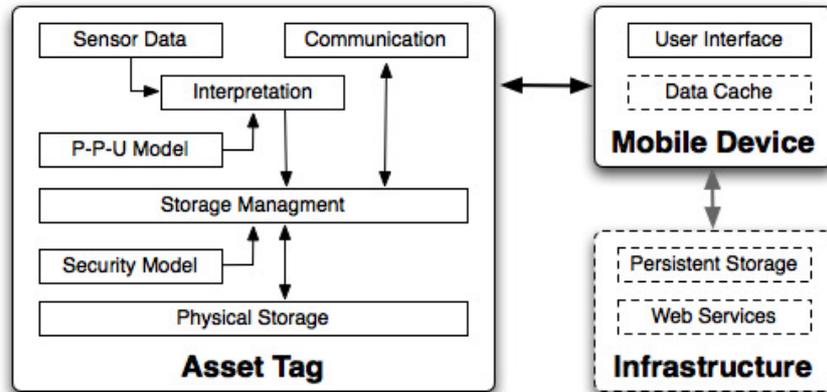
The function *Cost(m)* returns the cost for each type of mishandling event *m*, for the number of times the event occurs, *N*; this cost sums the base penalty for the occurrence of the event (returned by the *BasePenalty(m)* function) with a severity surcharge (returned by the *Severity(m)* function):

$$Cost(m) = \sum_{n=1}^N [BasePenalty(m) + Severity(m)]_n \quad (2)$$

In order to capture pay-per-use data we use wireless sensor nodes attached to the exterior of equipment. Equipment used in our construction scenario (as shown in Figure 1) exhibits vibration between certain ranges when in use (dependent on the individual piece of equipment). The sensor nodes contain an accelerometer and this is used to detect vibrations that indicate usage. The accelerometer is also used to detect experiences that involve sudden motions such as dropping (and the severity of the drop). The sensor nodes can potentially be used to detect other parameters in order to infer experiences such as monitoring temperature in order to detect overheating experiences.

### 3. Pay-per-Use Smart Object Architecture

As discussed in the previous section, our approach is to attach wireless sensor nodes to equipment which detect experiences such as usage. The self-contained sensor nodes maintain a semi-persistent memory of their experiences without reliance on other nodes or additional infrastructure. This independence is necessary to simplify deployment and ensure reliability in the context of remote building sites with limited connectivity and ad-hoc movement and usage of equipment. However, a backend database for persistent storage of experiences does exist and the use of mobile devices to interact with objects provides a form of opportunistic connectivity for transferring data. We assume that all users which interact with the smart objects have personal mobile devices such as mobile phones or PDAs, but these may have varying communication, storage and user interaction capabilities.



**Fig. 3.** Architecture of Prototype System Implementation

The overall architecture of the system is shown in Figure 3, where each piece of equipment is augmented with a wireless sensor node (labelled Asset Tag) which provides sensing, storage and processing. Mobile devices then interact with a sensor node (shown by the black arrow) and enable data to be transferred to a backend infrastructure (the dashed areas indicate future work). In summary, the sensor node interprets low-level sensor data to detect higher-level experiences of interest (e.g. equipment in use). Before the experiences are committed to physical storage, security and pay-per-use models are applied. Additionally, use of physical storage is carefully managed to ensure optimum use of space available and flash RAM (with a finite number of reads and writes). The mobile device is allowed to access data stored on the sensor node, subject to the security model. It is our intention that each user should be presented with their own personalised user interface based on the information they can access, the tasks which they are able to carry out and the capabilities of their mobile device. We now discuss the design of the security, communication and pay-per-use model aspects of the architecture in more detail.

While the key requirement for implementing the pay-per-use prototype is the detection and storage of usage experiences, we also wish to record involvement in processes that cannot be sensed (such as delivery) about which equipment has to be 'told' through user interaction. Storing a wider variety of information on a piece of equipment that is passed among different users in different companies raises the need for a security model. For example, the lessor is unlikely to allow the lessee to access information related to the profit generated through the lease of a piece of equipment. Conversely, the lessee may have to carefully control access information pertaining to individual workers use of the equipment.

Information stored at a piece of equipment may have been authored by the equipment itself (for example, interpreted from sensor readings), the lessor or the lessee. In order to provide data security, the categories of users or specific user that can access information are explicitly specified. In Table 1 we classify some example pieces of information which are stored on a sensor node in our prototype system according to the author and security restrictions.

Devices attached to equipment in the field have potential for only limited, opportunistic network connectivity with external back-end infrastructure. Therefore, embedded storage is essential for recording and accessing experiences that occur in real time. This requirement presents additional challenges for managing data on a device with limited storage capacity. However, the mobile phone used for interaction may also transfer data from the equipment to a backend infrastructure via network technologies such as GPRS/UMTS and also when synchronising the mobile device with a desktop machine.

**Table 1.** Examples of Data Stored on the Pay-Per-Use Prototype.

Author	Information	User Access
Sensor Node	Total use for current lessee	Lessor, Lessee: Administrator, Supervisor
	Per-worker usage for current lessee	Lessee: Supervisor, Worker (to which the usage is attributed)
	Maintenance experiences	Public
	Exception experiences for current lessee	Public
Lessor	Contact details	Public
	Pay-Per-Use billing model details	Lessor
	Serviceable lifetime of equipment remaining	Lessor
Lessee	Lease contract details	Lessee: Administrator, Supervisor
	Workers assigned to equipment use	Lessee: Supervisor, Worker

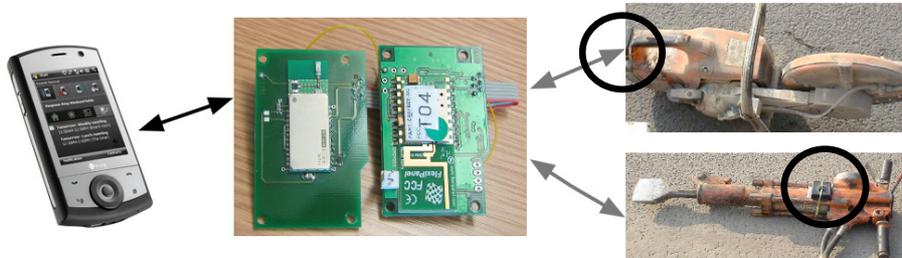
As discussed in section 2.4, a wide variety of pay-per-use billing calculations are possible which, in addition to usage time, could taking into account other parameters such as usage intensity and usage location. In our prototype scenario we currently implement the billing calculation at the piece of equipment in order to ensure accurate cost information is always available when interacting. However, we also assume that verifiable usage data retrieved during user interaction during and/or at the end of lease contract is used by the lessor to generate formal invoices.

## 4 Prototype Implementation

### 4.1 Hardware

The current prototype implementation is based on an existing hardware platform (Nemo Tags [6]) with wireless sensor nodes that include an accelerometer, ARM processor and low-power 802.15.4 radio enabling communication with and proximity detection. The platform is designed to enable HAV monitoring for extended periods in the field and consists of an Asset Tag which is fixed to equipment. Asset Tags are housed in a rugged enclosure (as shown in Figure 5) to deal with the harsh environment of building sites. They remain continually powered on (power

management is a key consideration [6] [7]) and automatically detect use without any user intervention.



**Fig. 5.** Overall Architecture Showing Mobile Phone Interaction with Equipment (Nemo Asset Tags Circled)

In order to enable mobile phone interaction with the Nemo Asset Tags a Bluetooth extension board was developed in order to provide a Bluetooth Bridge Tag. This enables communication between Nemo Asset Tags using 802.15.4 and Bluetooth enabled mobile devices. As interaction with equipment will take place in the harsh environment of construction sites, it was important to support a mobile device that was available in ruggedized format. Only a limited range of ruggedized phones that support development of 3rd party applications are available and as the majority of these are Windows Mobile devices this platform was chosen. Figure 5 shows the overall architecture for the prototype scenario where the black arrow represents Bluetooth communication and the grey arrow 802.15.4.

## 4.2 Software

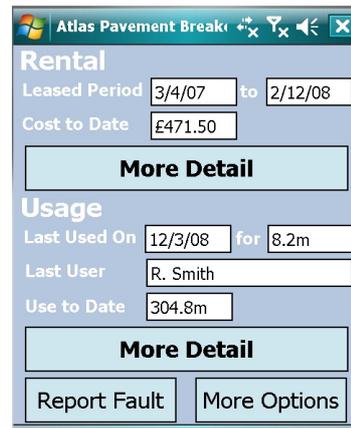
The applications that run on the Bluetooth Bridge Tag and Asset Tags attached to equipment are developed in C. The Bridge Tag continually listens for commands sent by a phone, validates those received and sends messages over the 802.15.4 radio as appropriate. A simple textual command set is currently used for all communication. Data received over the 802.15.4 radio is checked for integrity and forwarded to the phone if appropriate.

The mobile phone application, currently implemented in C#, initially discovers Bluetooth devices in range and allows the user to connect to the desired Bridge Tag. Once paired, communication takes place using Bluetooth SPP (Serial Port Profile) and commands are sent between the phone and bridge. An Asset Tag discovery process is initiated from the Bridge Tag (broadcast over the 802.15.4 radio) and tags in range reply with basic information such as equipment name, leaser name and status. The results of the discovery are used to populate a GUI on the phone application (Figure 6). Once the user has identified a piece of equipment they wish to interact with, they then 'connect' to it and access information via a customised user interface (currently implemented with a different application for each group of users). The interface for a supervisor in a construction company is show in Figure 7.

This early prototype implementation of the application and user interfaces is based on our analysis of the scenario and technical feasibility of features. We plan to refine our design through user studies with workers involved with equipment hire and use in the construction domain.



**Fig. 6.** Phone Interface Showing Results of Discovery Process.



**Fig. 7.** Construction Company Supervisor User Interface.

## 5 Evaluation and Discussion

### 5.1 Quantitative Evaluation of Storage Model

A fundamental requirement for pay-per-use implementation is persistent storage of usage experiences. Fulfilling this requirement in a resource-constrained embedded implementation requires careful management of available storage. From our analysis of usage data generated by a pavement breaker we found it possible to optimize the storage model.

From observations made during a field trial, typical use of a pavement breaker involves approximately 10 short bursts of drilling during a minute of use. The pavement breaker may be in continual use for upwards of 10 minutes in total (the period). Rather than logging raw data from the accelerometer, every continual burst of use is detected as a single usage experience. Assuming the record structure shown in Table 2 is used to store usage experiences where 20 bytes of storage are required to store every individual burst of drilling. The existing Nemo Asset Tag hardware implementation provides 460,800 bytes of available flash memory, which can potentially hold a maximum of 2304 minutes or 38 hours of use (assuming no other data is stored).

**Table 2.** A Record Structure for Storing Usage Events.

User (8 bytes)	Time Start (8 bytes)	Use Duration (4 bytes)
Unique User Tag ID	Seconds since epoch	Total in seconds

This storage limit may be exceeded when considering a piece of equipment that may be used heavily in the field over a lease period of several years. The solution we have to this problem is to apply a form of simple data aggregation (also called data fusion) [8] to aggregate the individual usage experiences during the overall period of drilling (10 minutes or more) into a single experience. This yields a minimum hundred-fold increase in storage capacity. Aggregating the data is effectively trading-off detail in the resolution of the data collected for overall storage capacity. However, the trade-off here is acceptable as the finest granularity of information we wish to capture is individual periods of drilling.

## 5.2 Evaluating the Trace of a Pay-Per-Use Example

We now consider the trace of experiences generated during a pay-per-use rental example and demonstrate the calculation of total cost for the customer. We consider the use of a pavement breaker over the course of five days with usage most days (shown in Table 3) and a single mishandling event (shown in Table 4). We detect use by comparing readings from the onboard accelerometer with known values which indicate use (recorded during testing in field trials), mishandling events such as dropping are detected in a similar manner.

**Table 3.** Usage Experience Records.

User	Time Start	Use Duration
9458001000000007	1212312600	429
9458001000000007	1212318907	365
9458001000000009	1212329054	671
9458001000000032	1212487974	967
9458001000000007	1212657043	243
9458001000000007	1212674247	401

**Table 4.** Mishandling Experience Record.

Time	Type	Severity Level
1212695901	1	12

Using the calculations presented in 2.4 we now find the total cost for the usage represented in tables 2 and 4. Using a per-second usage cost of £0.02, the total period of use in Table 3 (3076 seconds) the total usage cost is £74.12. Table 4 shows a single mishandling event, assuming *BasePenalty(1)* returns a value of £3 and *Severity(12)*

returns a value of £5 the total mishandling charge is £8. The total cost for the lease period is £82.12 and the total amount of storage required for the records shown in Tables 3 and 4 is 134 bytes. This realistic example highlights the feasibility of our pay-per-use design and is easily accommodated by our prototype system.

## **5.2 Discussion and Future Work**

Several issues remain to be addressed in the prototype implementation. In order to provide security in a scenario with groups of users with different privileges to access data (lessor, lessee etc.) we currently intend to implement this using encryption. The 802.14.5 radio module used on the Asset Tags [9] contains a hardware implementation of AES (Advanced Encryption Standard, also known as Rijndael) which enables the encryption of data in hardware during transmission, potentially removing the need to encrypt data stored on an Asset Tag. An important aspect of accurate pay-per-use billing is maintaining the integrity of usage data collected and stored on an object and this could be achieved through additional encryption (or even through the use of digital signatures and a trusted third party). Moving data from a pay-per-use object in the field to a back-end infrastructure (without resorting to the deployment of additional communication infrastructure such as that described [10]) is an area we wish to explore in the future. We consider that opportunistic connectivity provided by user interactions with objects from mobile phones is an excellent opportunity for this. However, challenges are raised if an object in the field cannot communicate with a back-end infrastructure directly and data has to travel via multiple asynchronous 'hops'. For example, an object would require confirmation that a piece of data has been successfully transferred to a back-end database intact before removing it from its own storage.

The use of a Windows Mobile device enabled rapid prototyping of user interfaces and interaction with the Nemo Asset Tags, in addition to the potential provision of a ruggedized user device. However, these devices require user interaction via a small keypad or keyboard and touch screen using a stylus. The use of these interaction methods, while not exclusive to Windows Mobile devices, require a very high degree of attention on the users part which may prove problematic on a building site where a user must be continually aware of their surroundings to avoid injury. A user may also be required to wear safety equipment on a construction site such as eye, ear and hand protection which may further increase the difficulty of interacting with a mobile device. It may be possible to help mitigate these problems through HCI design but an important aspect of future work will be discussion with industrial partners in order to explore these pragmatic user interaction challenges.

Refining and expanding the design of functionality provided by the prototype is a broader goal for future work. We plan to consider this extension in terms of what is technically feasible, what is desirable by users and how to support adoption of the system as part of users existing working patterns. We wish to explore these requirements through discussion of current and potential features with industrial partners using design workshops and other user-centred techniques.

## 6 Related Work

The idea of pay-per-use is traditionally applied to utility services (gas, electricity, phone, broadband etc.) but has recently been used in more dynamic automotive applications such as road pricing, where charges levied for the use of roads are also intended to reduce traffic congestion (the key concepts and examples of road pricing system are discussed in [11]). The concept has also been applied to car insurance, such as Norwich Union's 'pay-as-you-drive' product [1], where the customer attaches a device to their car that records use of the car and sends it to the insurer without user intervention. The Smart Tachograph [12] is a prototype system which takes this idea further and combines both road pricing and pay-per-use insurance. Smart Tachograph is similar to the work presented here in that it exploits the technology in use to provide more than purely pay-per-use functionality (such as providing a user interface). When inferring collecting car usage information in the form of GPS data (used in the 'pay-as-you-drive' product and Smart Tachograph) privacy issues emerge when disclosing data. GPS data can be analysed to determine factors such as whether the car exceeded speed limits and severity of acceleration/braking: information which can be used to determine whether a driver presents a high risk and should be charged more for their insurance. This potentially negative aspect of pay-per-use for certain users may emerge through our work. For example, through implementing pay-per-use some construction companies may find that they are paying more for hire of equipment than with a previous static scheme even though the extra charge is justified.

Related work describing existing prototypes of pay-per-use applied to objects is currently limited to a brief discussion of a pay-per-use chair idea [3]. However, the technological and financial motivations for implementing the prototype is similar to the work presented here. Several examples of prototypes that enable interactions with the environment via mobile devices exist, many use RFID tags and phones with in-built RFID tag readers. For example, the PERCI (PERvasive serviCe Interaction) architecture [13] enables interaction between the user and a set of semantic web services which compose the interactive aspects of the environment. In common with the aims of this work, the PERCI architecture supports the generation of user interfaces based on the capabilities of the user's mobile device. However, a phone equipped with an RFID tag reader is required and continual communication with web services using GPRS. Existing research has investigated a range of techniques for using mobile devices to interact with the environment which include touching, pointing, scanning and user mediated [14]. The technique of 'touching' is exemplified by RFID where a user has to physically touch an object to interact and 'scanning' describes interaction mechanisms such as Bluetooth where a user has to scan for objects in their current proximity. Pointing is typically implemented using a camera phone which recognises graphical markers located on an object and user mediated interaction involves a users specifying the object with which they wish to interact using an identifier such as a URL. Scanning is currently the most suitable method of interaction in our scenario as the user will be in close proximity of an object but physical contact may not be possible.

There exists extensive research involving wireless sensor networks (WSNs) in a variety of areas [15] [16]. Where large volumes of data are generated at nodes or

support for queries is required often a distributed query processing system such as tinyDB [17] is used. However, the application of this research is largely focussed on utilising data collected from communication between numerous (highly resource constrained) nodes to monitor an environment. While the application of sensor nodes may differ in WSNs several examples of hardware platform used (such as BTnodes [18]) are similar to that used in this work.

## 7 Conclusion

This paper has presented our initial exploration of the design, implementation and evaluation of a prototypical system for enabling pay-per-use. The approach used was to augment physical objects with wireless sensor nodes to enable them to become smart objects. The scenario used to motivate the design of our prototype is based on equipment rented for use in the construction domain. Once augmented with the prototype sensor nodes, equipment detects and collects its own experiences (such as usage and mishandling) then applies a pay-per-use billing model to calculate cost. We also support interaction with the smart objects via mobile devices from a range of users involved the equipment hire and usage scenario. These interactions are not restricted to pay-per-use applications, for example an administrator responsible for equipment procurement may interact to view usage data in order to revise hire decisions and a site supervisor may interact to report a fault with the piece of equipment. While the design and implementation of the prototype is at an early stage, a range of technical challenges have been identified and considered. These challenges included supporting user interaction with a smart object, data security issues in the context of multiple users and the need for managing the storage of experiences in an independent resource constrained sensor node. In addition to further refinement of design and evaluation of user interfaces, several additional issues were identified for future work. These included enabling user interaction with mobile device in an environment where a user must be continually aware of their surroundings to avoid injury, and transferring data to a backend infrastructure using opportunistic connectivity via mobile devices. Pay-per-use systems such as our prototype enable a wide variety of new and novel applications and we hope the work presented here will be valuable to other researchers exploring this area.

## References

1. Norwich Union Pay As You Drive Insurance, <http://www.norwichunion.com/pay-as-you-drive>.
2. MyRate Program, <http://auto.progressive.com/progressive-car-insurance/myrate-default.aspx>
3. Accenture Technology Labs, “Pay-per-use Object”, [http://www.accenture.com/Global/Services/Accenture\\_Technology\\_Labs/R\\_and\\_I/PayPerUseObject.htm](http://www.accenture.com/Global/Services/Accenture_Technology_Labs/R_and_I/PayPerUseObject.htm)
4. Kortuem, G., Alford, D., Ball, L., Busby, J., Davies, N., Efstratiou, C., Finney, J., White, M. I., Kinder, K.: Sensor Networks or Smart Artifacts? An Exploration of Organizational Issues

- of An Industrial Health and Safety Monitoring System. In Ninth International Conference on Ubiquitous Computing (UbiComp 2007), Innsbruck, Austria. Sept 16-19 (2007)
5. Davies, N., Efstratiou, C., Finney, J., Hooper, R., Kortuem, G., Lowton, M.: Sensing Danger – Challenges in Supporting Health and Safety Compliance in the Field. In 8th IEEE Workshop on Mobile Computing Systems and Applications (HotMobile 2007), Tucson, Arizona, February 26-27 (2007)
  6. Efstratiou, C., Davies, N., Kortuem, G., Finney, J., Hooper, R., Lowton, M.: Experiences of Designing and Deploying Intelligent Sensor Nodes to Monitor Hand-Arm Vibrations in the Field. In: ACM MobiSys 2007, San Juan, Puerto Rico, pp. 127–138, ACM Press. (2007).
  7. Brown, J., Finney, F., Efstratiou, C., Green, B., Davies, N., Lowton, M., Kortuem, G.: Network Interrupts: Supporting Delay Sensitive Applications in Low Power Wireless Control Networks. In Proceedings ACM MobiCom 2007 Workshop on Challenged Networks (CHANTS2007), Montreal, Canada, Sept 14 (2007)
  8. Heinzelman, W. R., Chandrakasan, A., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: 33rd Hawaii international Conference on System Sciences, Volume 8, IEEE Press, Washington, DC (2000).
  9. Chipcon 2004 CC2420 datasheet, [http://www.chipcon.com/files/CC2420\\_Data\\_Sheet\\_1\\_3.pdf](http://www.chipcon.com/files/CC2420_Data_Sheet_1_3.pdf)
  10. Kirisci, P.T., Hunecke, H.-H., Hribernik, K.A. Dikici, C.: A wireless solution for mobile collaboration on construction sites. In: IEEE International Workshop on Wireless Ad-Hoc Networks, Oulu, Finland, pp. 166- 171, IEEE Press (2004).
  11. Kelly, F.: Road pricing. <http://www.ingenia.org.uk/ingenia/issues/issue29/kelly.pdf> Ingenia 29 pp 34-40. (2006)
  12. Coroama, V.: The Smart Tachograph - Individual Accounting of Traffic Costs and Its Implications. In: Pervasive Computing, 4th International Conference, pp 135—152, LNCS, vol. 3968, Springer, Heidelberg (2006)
  13. Rukzio, E., Paolucci, M., Wagner, M., Berndt, H., Hamard, J., Schmidt, A.: Mobile service interaction with the web of things. In: 13th International Conference on Telecommunications, Funchal, Madeira island, Portugal, May 9-12 (2006).
  14. Rukzio, E., Broll, G., Leichtenstern, K., Schmidt, A.: Mobile interaction with the real world: An evaluation and comparison of physical mobile interaction techniques. In: European Conference on Ambient Intelligence, Darmstadt, Germany, November 7-10 (2007).
  15. Akyildiz, I.F., Su, W., Sankarasubramanisam, Y., Cayirci, E. Wireless sensor networks: a survey, In: Computer Networks 38 (4), pp. 393–422 (2002)
  16. Romer, K., Mattern, F., The Design Space of Wireless Sensor Networks, In: IEEE Wireless Communications, pp. 54-61 (2004).
  17. Madden, S. R., Franklin, M. J., Hellerstein, J. M., Hong, W., The design of an acquisitional query processor for sensor networks, In: 22nd ACM SIGMOD International Conference on Management of Data, pp. 491–502, (2003).
  18. Beutel, J., Kasten, O., Mattern, F., Romer, K., Siegemund, F., Thiele, L., Prototyping wireless sensor network applications with BTnodes. In: 1st European Workshop on Sensor Networks, volume 2920 of Lecture Notes in Computer Science, pages 323–338. Springer, Berlin (2004).