# SixthSense: RFID-based Enterprise Intelligence

Lenin Ravindranath$^{\dagger}$, Venkata N. Padmanabhan$^{\dagger}$, Piyush Agrawal$^{\ddagger *}$
$^{\dagger}$Microsoft Research India, $^{\ddagger}$Indian Institute of Technology Kanpur

## ABSTRACT

RFID is widely used to track the movement of goods through a supply chain. In this paper, we extend the domain of RFID by presenting SixthSense, a platform for RFID-based enterprise intelligence systems. We consider an enterprise setting where people (or rather their employee badges) and their personal objects such as books and mobiles are tagged with cheap, passive RFID tags, and there is good coverage of RFID readers in the workplace. SixthSense combines mobility information obtained from RFID-based sensing with information from enterprise systems such as calendar and presence, to automatically draw inferences about the association and interaction amongst people, objects, and workspaces. For instance, SixthSense is able to automatically distinguish between people and objects, learn the identities of people, and infer the ownership of objects by people.

We characterize the performance of a state-of-the-art RFID system used in our testbed, present our inference algorithms, and evaluate these both in a small testbed and via simulations. We also present the SixthSense programming model that exposes a rich API to applications. To demonstrate the capabilities of the SixthSense platform, we present a few applications built using these APIs, including a misplaced object alert service, an enhanced calendar service, and rich annotation of video with physical events. We also discuss the issue of safeguarding user privacy in the context of SixthSense.

**Categories and Subject Descriptors:**
C.m [Miscellaneous]: Sensing Systems

**General Terms:**
Design, Experimentation, Measurement.

**Keywords:**
RFID, sensing, ubiquitous computing.

## 1. INTRODUCTION

Radio Frequency Identification (RFID) [15, 27, 13] is an electronic tagging technology that allows the detection and tracking of tags, and consequently the objects they are affixed to. An RFID tag typically comprises a passive transponder that responds with identifying information when energized remotely by an RFID reader. This ability to do remote detection and tracking coupled with the low cost of passive tags has led to the widespread adoption of RFID in the supply chain world. RFID is used to track the movement of goods through a supply chain, whether it be pallets shipped between warehouses, cases delivered to stores, or items placed on the store shelves, thereby optimizing inventory management and yielding significant cost savings.

The promise of cheap "connectivity" to any object carrying an RFID tag has led to the vision of an "Internet of Things" [11, 26]. Our work on SixthSense is inspired by this vision. SixthSense focuses on applying RFID to an enterprise setting, such as a corporate office or university department. An enterprise setting is different from the supply chain scenario in one fundamental way: the central role of people. Unlike in a supply chain, an enterprise setting involves rich interaction amongst people, between people and objects, and between people and workspaces. For instance, people own objects such as books, cell phones, and laptops, which they often carry around and sometimes misplace. SixthSense provides a platform for tracking and inferring such interactions, and then exposing these to the higher layers via APIs that enable useful applications and services to be built. Thus SixthSense raises the level of abstraction for applications in this domain beyond tag-level events, akin to how RFID stacks such as Microsoft's BizTalk [7] do so in the supply chain context. In short, SixthSense represents a form of *mobile computing applied to non-computing entities.*

SixthSense assumes a setting where most people (or rather their employee badges) and objects are tagged with passive RFID tags, and the coverage of RFID readers spans much of the workspace. However, we do *not* assume that this tagging is always catalogued systematically. Indeed, many objects present in a workplace may not even belong to the enterprise (e.g., a user's personal mobile phone). Even if all objects (and people) were cataloged, this would be a manual process prone to errors and furthermore would require updating each time a new object is added or an object needs to be retagged because of the deterioration of its old tag [8]. Therefore, a key goal of SixthSense is to make all inferences automatically, without requiring any human input. Even in settings where human input is available, the inference algo-

[*] The author was an intern at Microsoft Research India during the course of this work.

*MobiSys'08,* June 17–20, 2008, Breckenridge, Colorado, USA.
Copyright 2008 ACM 978-1-60558-139-2/08/06 ...$5.00.

rithms in SixthSense can help catch errors, e.g., the wrong ownership information for an object being recorded in a catalog.

SixthSense incorporates algorithms that start with a mass of undifferentiated tags and automatically infer a range of information based on an accumulation of observations. SixthSense is able to automatically differentiate between people tags and object tags, learn the identities of people, infer the ownership of objects by people, learn the nature different zones in a workspace (e.g., private office versus conference room), and perform other such inferences. *Mobility of people and objects is key to the inference performed by SixthSense.* For example, tags attached to people are more likely to move, with less dependence on other tags, than tags attached to objects. Likewise, the owner of an object is likely to be the person who carries it around the most.

Since RFID by itself only provides very limited information — basically, just the presence or absence of a tag in a particular zone — *SixthSense also leverages information from other enterprise systems*, e.g., calendar, presence, login information, etc. By combining information from these diverse sources, SixthSense records all tag-level events in a *raw* database. The inference algorithms consume these raw events to infer events at the level of people, objects, and workspace zones, which are then recorded in a separate *processed* database. Applications can either poll these databases (e.g., by running SQL queries) or set up triggers to be notified of specific events of interest. We present a few applications that we have implemented on top of SixthSense: lost object alert, enhanced calendar and presence, semi-automated image cataloging of objects, and rich annotation of video with physical events.

We envision SixthSense being run centrally by the enterprise rather than by individual users. This includes the raw and processed databases, and the applications that consume the information contained in these databases. While such a model limits flexibility, it greatly simplifies deployment issues, specifically with regard to privacy. Individual user's are unable to access the SixthSense databases; they are only presented with information that the centrally-run application chooses to expose, e.g., alerts regarding a user's own objects that have been misplaced. We also employ a simple tag relabeling scheme to defeat any attempts to reconstruct the database surreptitiously, say using input from rogue readers. While the enterprise itself will have access to potentially privacy-sensitive data, this is not fundamentally different from the present situation, where the enterprise has access (with legal sanction, in some countries [17]) to arguably more sensitive information such as the employees' email and files (largely in an unencrypted form), and indeed also the ability to track user movements to an extent based on card key based access control (which is, in fact, based on short-range RFID) to various physical spaces. We believe that the safeguards in place to guard against leakage or abuse of such sensitive information could be extended to SixthSense.

In summary, the main contribution of our work is the design, implementation, and evaluation of SixthSense, a platform for RFID-based enterprise intelligence that combines RFID events with information from other enterprise systems and sensors to automatically make inferences about people, objects, workspaces, and their interaction.

| | | |
|---|---|---|
| Class 0 | Passive | Read only |
| Class 1 | Passive | Read only write once<br>but with rewritable 96-bit EPC |
| Class 2 | Passive | 65 KB read-write |
| Class 3 | Semi-passive | 65 KB read-write<br>with built-in battery |
| Class 4 | Active | Built-in battery |
| Class 5 | Active | Communicates with other<br>class 5 tags and devices |

**Table 1: Different classes of tags.**

## 2. RFID BACKGROUND

Radio Frequency Identification (RFID) [15, 27, 13] has been around for decades. It is generally believed that the roots of RFID can be traced back to World War II [10], when the British first put transmitters on their aircraft, which on receiving radar signals, broadcast back a signal to ground station identifying the plane as a friend. Since then, the technology has improved tremendously and RFID has seen large deployments, especially in the supply chain and asset tracking domains.

An RFID system comprises a reader, with one or more antennas attached to it, and tags. When a reader energizes an antenna, the tags in the corresponding zone get activated and respond with their ID (e.g., a 96-bit electronic product code (EPC)) and possibly other data.

RFID tags come in three types: *passive*, *active* and *semi-passive*. Passive tags do not have any internal power supply. Instead, they use the electric current induced in the tag's antenna by the incoming RF signal from a reader to power the tag's IC and transmit a response. Such tags typically have a read range of about 10 cm up to a few meters. The simplicity of these tags has also meant a low cost — about 15 U.S. cents per tag today, expected to go down to 5 U.S. cents [9]. Active tags, on the other hand, have their own internal battery to power the IC and transmit a response using an arbitrary RF technology such as WiFi. They have a read range of hundreds of meters, due to the internal battery. Semi-passive tags have their own power supply to power the IC and to help with reception, but like passive tags they use the RF induced current for transmitting a response back to the reader.

Passive tags typically receive power through inductive or radiative coupling. Inductive coupling is used for powering LF (low frequency, 30-300kHz) and HF (high frequency, 2-20 MHz) tags. Such tags receive power in the *near field*, which refers to the region within a few wavelengths of the reader's antenna. A reader antenna generates a magnetic field, inducing an electric current in the tag's antenna and charging a capacitor in the tag. Radiative coupling is used for UHF tags (Ultra High Frequency, above 100 MHz). In this case, the tag antenna receives signals and energy from the electromagnetic field emitted by the reader in the *far field*, the area beyond a few wavelengths.

With increasing RFID deployments, a need for standardization was felt for ensuring interoperability of the RFID systems from different vendors. The Auto-ID Center at MIT, which is now being managed by EPCglobal [1], developed the electronic product code (EPC). EPCglobal has defined different classes of tags, as shown in Table 2.

Class-0 and Class-1 tags are not interoperable and they are not compatible with ISO standards. In 2004, EPCglobal began developing a Class-1 Generation-2 protocol (Class-1 Gen-2 or just Gen-2), which would not be backward compatible with either Class-1 Generation-1 (Class-1 Gen-1) or Class-0 tags. The aim was to create a single, global standard that would be more closely aligned with ISO standards. Class-1 Gen-2 was approved in December 2004.

With the increasing use of RFID technology for retail systems, there has been a concern that the privacy of individuals that purchase items would be jeopardized by the ability to identify items uniquely and surreptitiously. Responding to this concern, the Auto-ID center, and later EPCglobal, included an option to kill a tag after purchase if a customer desires to protect their privacy. For example, in Class-0 tags, a 24-bit password is programmed into the tag during manufacturing. The password can be used to kill the tag. Once a reader accesses a tag with the correct 24-bit password and issues the kill command, a fusible link on the tag becomes open, rendering it unreadable.

## 3. EXPERIMENTAL SETUP

We briefly discuss our experimental setup.

### 3.1 RFID Equipment

We used an Impinj Speedway reader [4] for our experiments. This is a state-of-the-art UHF Class 1 Gen 2 reader that is compliant with EPCglobal and ISO standards. It operates in the 865-956 MHz band. Our setup is equipped with 4 patch antennas, each measuring 26 cm by 26 cm. The RF power output by the antennas is +30 dBm (1 W). The receiver sensitivity of the reader is -80 dBm.

We tag objects with Impinj Monza passive RFID tags [3]. These are UHF Class 1 Gen 2 tags that have been certified by EPCglobal. It includes a 96-bit field-rewritable EPC and supports a 32-bit password-protected kill command. The tag, including the chip and packaging, measures 9 cm by 5 cm.

The reader connects to the corporate network using Ethernet and the host computer connects to the reader using its IP Address. The reader can be connected to 4 RFID antennas. The Speedway reader exposes the RSSI of a tag being read in addition to its EPC. The reader also has the capability to write tags.

Figure 1 shows a picture of the reader with its antennas and a few objects that were tagged.

### 3.2 Physical Setup and Enterprise Setting

Our work is set in the Microsoft Research India Lab. We deployed our RFID reader to cover a section of the workspace on one floor measuring 10 meters by 6 meters. This space is occupied by 4 users, who served as the test subjects for our experiments. We tagged several objects belonging to each user: their employee badges, mobile phones, laptops, books, water bottles, etc. These users used a calendar system hosted on Microsoft's corporate Exchange servers. Also, they logged in to a corporate domain and signed in to a Microsoft Universal Communicator presence service, both of which indicated user activity at their computer. [1]

Figure 2 shows a view of the workspace, including the RFID reader antennas and the users along with their tagged objects. While the antennas are placed on the users' desk in the current setup, our eventual plan is to have these mounted on the ceiling.

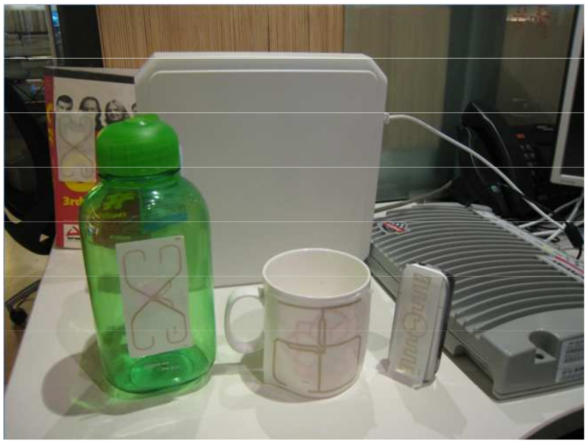


**Figure 1: Speedway Reader, Antenna and Monza Tags**

**Figure 2: RFID deployment**

### 3.3 SixthSense Simulator and Visualizer

Given the small size of our testbed, we developed a simulator to enable experiments at larger scale. The simulator incorporates simple models of object ownership, user mobility (possibly carrying one or more objects), objects being misplaced, user logins, etc. The simulator then generates a synthetic trace of RFID and other events, which is then fed into the SixthSense system for analysis and inference. Our tool also incorporates a visualizer that depicts users, objects, and their movements.

## 4. RFID PERFORMANCE MEASUREMENTS

While there have been many measurement studies of RFID, as discussed in Section 11, we would like to characterize the performance of the Impinj Speedway reader in our particular setting. To this end, we present some basic performance measurements.

[1] We would want to filter out login or sign-in activity performed remotely, which would be straightforward to do.

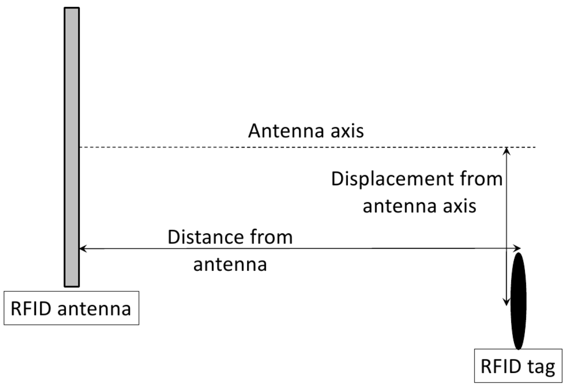


**Figure 3: Measurement setup**

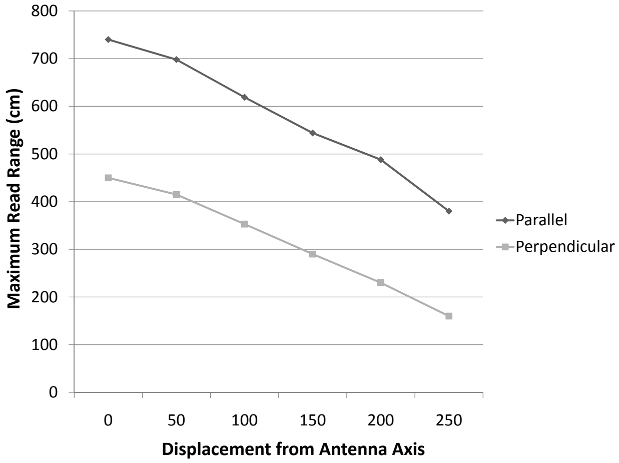


**Figure 4: Effect of displacement from antenna axis on read range**

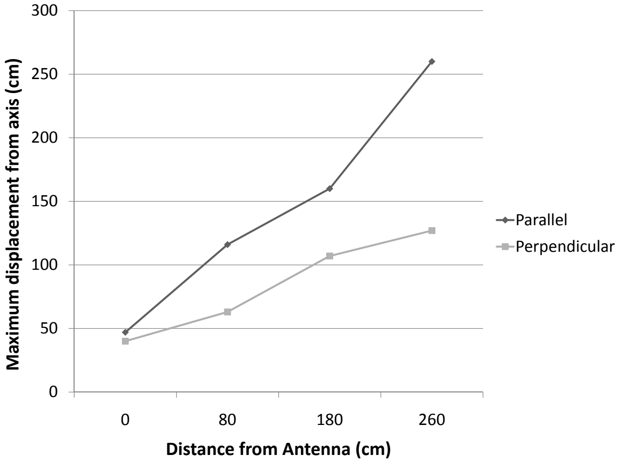


**Figure 5: Effect of distance from antenna on allowed displacement from axis**

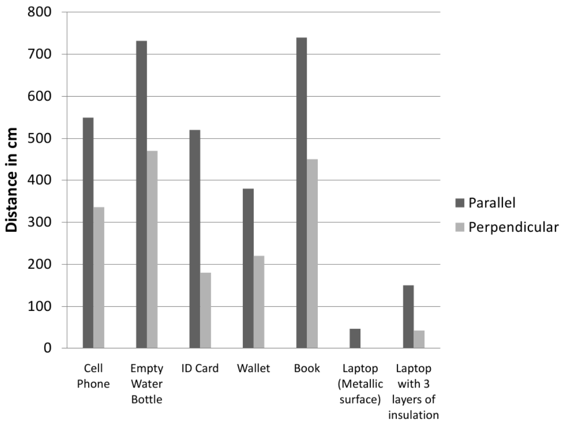


**Figure 6: Read ranges for tags on objects carried by a person**

In the first set of experiments, we studied the attainable read range for different displacements of the tag from the normal axis of the antenna. Figure 3 shows the measurement setup. At each displacement from the antenna axis, the maximum read range is recorded as the farthest distance at which the tag can still be read. It can be inferred from Figure 4 that as the tag moves away from the antenna axis, the read range decreases. Also, when a tag is in a perpendicular orientation with respect to the plane of the antenna, the read range is lower. In another experiment, we kept the tag at different distances from antenna and measured the maximum allowable displacement from the antenna axis that still allowed the tag to be read. Figure 5 shows that as the tag moves farther from antenna, the allowed displacement from axis also increases proportionally. Also, a tag in perpendicular orientation allows a much smaller displacement from axis as compared to a tag in parallel orientation.

We also noted that when a tag is in the range of the reader, the read rate or response rate (i.e., percentage of times a tag is read when probed) is 100% and when it is outside the range, the read rate immediately falls to 0%. This behavior is in sharp contrast to the more gradual degradation in read rate at the read range boundary, as reported in other studies [22, 23]

We then characterized the read ranges for tags affixed to objects such as cell phone, wallet, books and laptops, which are often carried by people in an enterprise environment. Figure 6 shows the observed read ranges for different objects in both parallel and perpendicular orientations. Tags affixed to books have a read range similar to that of tags in the open, i.e., not affixed to any object. An interesting observation was the effect of insulation layers on tags attached to metallic bodies (e.g., laptops). We observed that without any insulation, such tags had very low read ranges (50 cm). However, with just 3 layers of thin card board insulation, we were able to obtain read ranges of 150 cm. In general, our experiments showed that the read range increases with an increasing number of insulation layers. It can also be noted from Figure 6 that the read range is lower when the tag is oriented perpendicular to the antenna as compared to when it is oriented parallel to the antenna.

Setting the transmission power of the reader to different values, we experimented with the read ranges obtained. We observed that as the power increases, the read range also increases. With the highest power level, the observed antenna range was about 750 cm (7.5 m), which is large enough to cover most rooms in an office building. This is the power setting we used in all of the SixthSense experiments.

# 5. SIXTHSENSE ARCHITECTURE

We now lay out the overall architectural rationale and structure of SixthSense.

## 5.1 Assumptions

SixthSense assumes an enterprise setting with widespread coverage of RFID readers, and where most or all people and objects are tagged with passive RFID tags. However, we do *not* assume that the tagging of people and objects is cataloged. Users are free to pick up new tags and affix them to objects, as and when needed. This low-overhead model with little control is appropriate for SixthSense because these long-range UHF tags do not serve any security function, unlike the short-range HF tags embedded in employee card keys.

We also assume that users have access to a computing environment that provides services such as network logins, shared calendars, and online presence, which can be monitored by SixthSense. This is increasingly the case in enterprises, with the adoption of networked systems such as Microsoft Exchange and IBM Lotus Notes.

## 5.2 Architectural Components

Figure 7 shows the SixthSense architecture. The key components of the system, including the databases, inference engine, and applications, are run centrally by the enterprise. This provides the (trusted) inference engine access to the complete set of sensed data across all users, objects, and zones, allowing it to make effective inferences. Likewise, the (trusted) application is allowed the flexibility of working with a complete set of inferences (i.e., inferences pertaining to *all* users and their objects), yet control what processed information is presented to the users to ensure privacy. In contrast, if the inference engine or the application were run by individual users on their own desktop machines, privacy consideration would restrict the set of information made available to these, and hence limit their functionality. For example, privacy considerations would disallow an application run by one user from accessing inferences pertaining to another user, making it difficult to implement new functionality such as the automatic conference room booking feature discussed in Section 9.3.

Next, we briefly discuss the various components of SixthSense.

### 5.2.1 RFID Monitor

The RFID Monitor issues a read command every 500 ms to the RFID reader. The reader reports the EPC and the signal strength (RSSI) of the tags read via each of its antennas. This data then gets pushed into the raw database.

### 5.2.2 Other Enterprise Monitors

These other monitors monitor the information listed below and push their updates into the the raw database:

- *Calendar Monitor*: This resides on each user's desktop machine, and monitors the time and location of the user's appointments.

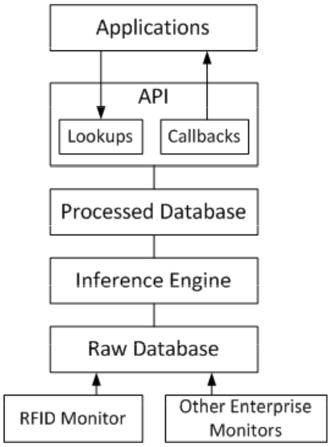


**Figure 7: SixthSense Architecture**

- *Presence Monitor:* This monitors the status of each user's interaction with their desktop. A machine is said to be idle when it receives no user input for 2 minutes. Transitions from idle to active state are detected and reported.

- *Login Monitor:* This is similar to the presence monitor except that in general login is a much stronger indication of a user being present than simply a change in their machine's idle.

- *Cameras:* Office buildings are often equipped with cameras for security reasons. The camera feed is stored in a video database for future analysis, if the need arises. We show in 9.2 how we combine the camera system in an enterprise with other sensors to build useful applications.

### 5.2.3 Raw database

The RFID monitor and the other enterprise monitors push data into the raw database.

### 5.2.4 Inference engine

The inference engine operates on the raw database using the algorithms explained in Section 6 to draw inferences about people, objects, and workspaces.

### 5.2.5 Processed database

The processed database is populated by the inference engine with its inferences, making these available to applications built on top of the SixthSense platform.

### 5.2.6 API

SixthSense provides a set of APIs for applications to lookup the inferences stored in the processed database or to receive callbacks when new inferences are made. We elaborate on this in our discussion of the SixthSense programming model in Section 8.

### 5.2.7 Applications

A range of applications can be built using the APIs exposed by the SixthSense platform. We discuss a few that we have built in Section 9.

# 6. ALGORITHMS

We discuss the algorithms used by the SixthSense inference engine to perform automatic inference and to address some systems challenges.

## 6.1 Automated Inference

The algorithms in SixthSense operate on a stream of events, both from the RFID readers and from other enterprise systems, to make several inferences automatically. We discuss these in sequence, from the basic ones to the derived ones.

For ease of exposition, we assume that all people and objects are tagged and that these tags are read reliably by the reader antenna that covers the zone they are in. However, we also discuss the impact of deviations from this ideal. Movement is quantified in terms of the number of inter-zone transitions. For instance, if a person takes a walk that takes them through 4 zones along the path $A \rightarrow B \rightarrow C \rightarrow D$, this would correspond to 3 units of movement.

Finally, much of our analysis is based on the movement of tags between zones, so we conceptually assume that the zones are non-overlapping. In practice, however, multiple zones may overlap (i.e., the tag is read by more than one antenna at the same time), but we effectively eliminate the overlap by filtering out all of the concurrent reads of a tag from multiple antennas except for the one with the strongest signal strength. To eliminate the possibility of incorrect inferences made because of spurious inter-zone movement, we could also disregard any movement between overlapping zones. However, we defer further investigation of this possibility to future work.

### 6.1.1 Person-Object Differentiation

The goal of person-object differentiation is to take an undifferentiated mass of tags and classify each tag as either belonging to a person or to an object. A "person tag" refers to a tag affixed, say, to a person's employee badge, which is (almost) always carried by him/her.

The essential difference between people and objects pertains to their mobility. People can move on their own whereas objects move only when carried by a person. To make this differentiation, we might be tempted to measure the amount of inter-zone movement exhibited by each tag and classify the ones with a high degree of movement as people and the rest as objects. Besides the problem of defining a suitable threshold on the degree of movement to decide whether a tag is a person tag, there is also the more basic problem that this heuristic could yield the wrong inference in many cases. For instance, an object belonging to an active person (e.g., his/her mobile phone) might well exhibit far more movement than a sedentary person.

To address this difficulty, we use a *co-movement* based heuristic. The basic idea is to consider the movement of a tag not in isolation but in relation to the movement of other tags it moves with. We say that two tags move together when they make the same inter-zone transition at about the same time. The movement of a tag is compared with the movement other tags, whether attached to objects or to people, that it moves with at some point. If on average the tag moves more than these other tags, it is deemed to be a person. Otherwise, it is classified as an object. So in the above example, the high degree of movement exhibited by an active person's mobile phone would be compared with the even higher degree of movement of the active person himself/herself. Consequently, the phone would *not* be misclassified as a person.

Formally, consider a tag $T$ that exhibits a total movement of $m$. (Recall that this corresponds to $m$ inter-zone transitions.) Let the set of tags that $T$ moves with at some point in time (i.e., makes a near-simultaneous inter-zone transition with at some point) be $CM_T = \{T_1, T_2, \cdots, T_n\}$. We refer to $CM_T$ as the *co-movement set* for $T$. Let $m_1, m_2, \cdots, m_n$ represent the amount of total movement (with or without $T$ being present) exhibited by $T_1, T_2, \cdots, T_n$, respectively, and $c_1, c_2, \cdots, c_n$ be the amount of co-movement exhibited by each of these tags together with $T$. Let $C = \sum_{i=1\cdots n} c_i$. Then we define the *relative movement* metric, $RM_T$, for tag $T$ as:

$$RM_T = m - \sum_{i=1\cdots n} \frac{c_i}{C} m_i$$

Intuitively, $RM_T$ is the amount by which $T$ moves minus the average movement exhibited by the tags it moves with. We weight the movement of each other tag by the amount of its co-movement with $T$ (as reflected in the $\frac{c_i}{C}$ factor) to avoid having occassional co-movements skew the metric. For instance, a sedentary person might occassionally move with a highly active person. However, we would not want this to obscure the much more frequent co-movements that the sedentary person exhibits with one or more for his/her objects.

We compute $RM_T$ for all tags $T$. Then we find the tag $T$ that has the largest (positive) $RM_T$ and declare it to be a person. We then adjust the co-movement sets for other tags as follows. Consider two other tags, say $T_1$ and $T_2$, that exhibit co-movement with $T$. To decide whether there are any instances of co-movement between $T_1$ and $T_2$, we first eliminate all instances where $T$ (which has already been declared to be a person tag) also moved. Only if $T_1$ and $T_2$ move together without co-movement by $T$ as well do we consider $T_2$ to be part of $T_1$'s co-movement set, and vice versa. The intution is that when there is already a person present to explain a movement, we should *not* use the *same* movement as evidence of another tag being a person. For instance, this would prevent a person's mobile phone, which is often carried by him/her, from itself being mistakenly classified as a person based on its apparent co-movement with a book that the person also carries on occassion.

Once the co-movement information has been adjusted as noted above, we recompute $RM_T$ for all of the remaining tags and repeat the above process so long as there remains a tag with a positive $RM_T$. Once we reach the point where $RM_T$ is negative for all of the remaining tags, we declare these to be object tags and terminate the algorithm.

Finally, tags that do not exhibit any movement are classified as objects. People are very unlikely to remain immobile for days or even hours at a stretch. On the other hand, an object such as a book may never move out of the zone corresponding to the owner's office.

### 6.1.2 Object Ownership

Having classified tags as people or objects, we turn to the question of inferring the owner (obviously a person) of each object. Although co-movement might correlate with ownership (i.e., an object is generally carried by its owner), it might fail in some not-so-uncommon situations. For instance, a person might refer to a book he/she owns only in his/her own office. The only time when the book moves across zones is when someone else borrows it. So a comovement based inference would likely get confused and attribute ownership of the book to the occassional borrower.

Instead, we use a simple *co-presence* based heuristic, which works as follows. For each object, we simply keep track of the amount of time that the object is concurrently present in the same zone as a person. The person with which an object is co-present the most is deemed to be the owner of the object. For instance, all of the objects that are owned by a person and that are generally in his/her office would be deemed to be owned by that person since it is he/she who would be spending the most amount of time in the same zone (i.e., his/her own office) and at the same time as the objects. While there is the possibility of misclassification (e.g., when an object that is borrowed by someone, who holds it for longer than the owner himself/herself had it), the likelihood of this is low.

There are also cases of objects that are not owned by anyone in particular, e.g., a book in a common lounge or a mug in a kitchen. The above heuristic would find that no one person dominates the co-presence metric for such an object, and this observation could be used to identify it as a *shared* object. However, we have not investigated this further as of this writing.

### 6.1.3 Zone Identification

An office building typically has different kinds of workspaces. There are workspaces assigned to individuals, be they private offices, semi-private cubicles, or desks. There are shared spaces, either reserved spaces such as meeting rooms or non-reserved spaces such as lounges or reading rooms. Finally, there are also common areas such as hallways and stairwells, which people generally move through rather than stay put in for significant lengths of time (occasional hallway conversations notwithstanding).

Although an enterprise may know the nature of each physical workspace, it would be an overhead to keep this in sync with the RFID deployment and the zone of coverage of the many RFID antennas (e.g., an antenna may be moved or pointed differently without the master database being updated). So in SixthSense, we automatically classify workspaces (actually, the RFID zones covering those workspaces) using information on the presence, movement, and calendar information of people (not objects), as follows:

- *Individual workspace:* If there is one person who is predominantly present in a zone, we deem that to be an individual workspace for that person.

- *Shared workspace:* If no one person is predominantly present in a zone and the mean residence time of a person in the zone (i.e., the length of time from when the person enters the zone to when they exit it) is greater than a threshold, we deem the zone to be a shared workspace. Even a threshold of just 5-10 minutes would be sufficient to differentiate such a space from a common area that people simply pass through.

- *Reserved shared workspace:* A shared space such that people who are present in the space for more than a threshold period often have common meeting entries in their calendars, with matching locations and times, is deemed to be a reserved shared workspace (e.g., a reserved meeting room).

- *Common areas:* Any space that is not classified as one of the above is deemed a common area.

### 6.1.4 Person Identification

Having identified the *nature* of tags (person vs. object) and the nature of workspaces, we now turn to inferring the *identity* of the person corresponding to a person tag. The basic idea is as follows. In a modern workplace, with computers at every desk, a user often interacts with their computer soon after entering their office (e.g., by logging in or causing their presence information to transition from “away” to “online”). So we can expect to find a correlation, even if not perfect, between the stream of events corresponding to users entering *their* individual workspaces (*entrance* events) and the same users interacting with their computers (*login* events). This can help identify the person corresponding to a tag.

To codify this intuition, we maintain a graph where person tags and person identities are the nodes, and the weight of the directed edge from a person tag to a person identity reflects our belief in their correspondance. When an entrance event corresponding to a person tag and the login event corresponding to a (possibly different) person happen within a short window of time, we increment the *coincidence count* for the corresponding edge. Note that many (entrance,login) events may happen within the same window, and so we may increment the *coincidence counts* for multiple edges. We also keep track of the total number of entrance events for each tag. The ratio of the *coincidence count* for a (tag,identity) pair to the total number of entrance events for the (person) tag yields the weight for the corresponding directed edge. The highest weight edge emanating from the node corresponding to a person tag would then point to the identity of the corresponding person. Note that we would need to gather a sufficient number of samples to avoid situations where multiple tags point to the same identity with their highest weight edges.

To help scale this algorithm, we would want to compartmentalize the entrance and login events, say based on geographic regions. For instance, in the Microsoft context, there is little point in mixing up login events that happen in a subnet in Bangalore with entrance events that happen in a zone in Redmond. We plan to investigate this refinement in future work.

### 6.1.5 Object Interaction

Finally, we turn to examining events *within* an RFID zone. Specifically, we consider the problem of inferring that an object has been “interacted” with, e.g., picked up by a person. Our approach is simple. Consider a set of tagged objects in a zone. If the environment is stable, the RSSI of the backscatter from the tags as recorded at the reader would also be stable. However, if an object is picked up, this would typically change its distance from and/or its orientation with respect to the reader antenna, thereby causing the RSSI of its tag to change.

Based on empirical observations, we use the following procedure to detect a significant change. We sample the RSSI of each object tag every 200 ms. Then, in a sliding 4-second wide window containing 20 samples, we determine the $10^{th}$ and $90^{th}$ percentiles of the RSSI. If these differ by at least 10 dBm, we conclude that there has been a significant change in the RSSI and declare that the object has been interacted with.

The RSSI could also change because of the introduction of an obstruction, e.g., the movement of a person. To minimize spurious detection of interaction because of such events, we could have multiple antennas, each mounted in a different position and orientation, covering the region of interest. We would then insist that each antenna detect a significant chance in the RSSI of a tag (based on the thresholds noted above) for it to be deemed as an interaction. The chances are low that a non-interaction event would cause the RSSI measured by all antennas to change significantly. As our results in Section 7.1.2 show, a conjunction of two antennas yields excellent results.

Note that prior work [16] has used variations in the response rate of tags to infer object interactions. However, as mentioned in Section 4, we find that there is little variation in the (100%) response rate when a tag is well within the range of a reader. Indeed, the range of our reader reported in Figure 4 is larger than the size of a typical individual office. Hence we find that using RSSI information as a more promising approach to detecting object interactions.

## 6.2 Systems Challenges

We now discuss a couple of key systems challenges that arise in the context of SixthSense: improving reliability and ensuring privacy.

## 6.3 Improving Reliability

To improve the reliability of RFID tag reads, we use a simple *multi-tagging* scheme, where we affix multiple tags, in different orientations, onto each object. The orientation diversity helps increase the probability of at least one of an object's tags being detected. This technique has been used with success in prior work [22].

However, in the context of SixthSense, where automation is key, multi-tagging raises the issue of automatically learning which *set* of tags is attached to the same object. We start with the assumption that all tags belong to one giant super-object, represented as a fully-connected graph over the tags. Any time two tags are detected simultaneously in different zones, we conclude that the tags belong to different objects and so delete the edge between them. (A refinement, for robustness, would be to insist that the different zones be non-adjacent.) After running a large number of tag sighting events through this algorithm, we would be left with a set of connected components, each of which we infer as corresponding to the set of tags attached to the same object.

## 6.4 Ensuring Privacy

As noted in Section 1, we do not worry about defending against privacy attacks by the enterprise itself, which is after all the entity that would deploy and manage a system such as SixthSense. Instead, we focus on attempts to compromise privacy by surreptitiously monitoring RFID tags, say using rogue readers. [2]

We use a simple relabeling technique where the ID (e.g., EPC code) on each tag is rewritten by the SixthSense infrastructure at random times, thereby defeating attempts by an attacker to track specific tags. This is akin to the relabeling technique discussed in [19] except that relabeling in SixthSense is a continual process rather than just a one-time step performed at the time of checking an item out from a retail store. Only the infrastructure would be aware of the mapping between the old and new tag IDs. Note that unless the relabeling process is password protected (which it is not with our RFID setup), a rogue reader can mount a DoS attack by randomly relabeling tags, unbeknownst to SixthSense. However, this still would not compromise privacy. Furthermore, if such rogue attacks are sporadic, the automated inference algorithms in SixthSense (Section 6.1) would be able to converge back to the correct inferences soon enough.

| Object | Reliability | |
|---|---|---|
| | (at 1 m) | (at 2 m) |
| Badge on belt clip | 100% | 96% |
| Small box in hand | 94% | 88% |

**Table 2: Detection of inter-zone movement.**

# 7. EVALUATION

We now present an evaluation of SixthSense's inference algorithms. We start with experimental results based on the testbed described in Section 3. This evaluation focuses primarily on inferences that are impacted by physical RFID effects. The small scale of our testbed, comprising one reader with four antennas, means that we need to turn to simulations to evaluate many of the inferences algorithms at scale.

## 7.1 Testbed-based Evaluation

We start with an evaluation inter-zone movement detection and object interaction detection. We then present some results from our small-scale deployment of SixthSense.

### *7.1.1 Inter-zone Movement Detection*

Several of the inference algorithms in SixthSense depend on the detection of inter-zone movement. To evaluate the reliability of inter-zone movement detection, we set up two RFID antennas along a wall and had the user walk past the antennas at a certain distance from the plane of the antennas. The experiment involved the user making 100 inter-zone crossings, while wearing their employee badge on their belt clip and carrying a small cardboard box in their hand. Each of the badge and the cardboard box was tagged.

Table 2 reports the fraction of inter-zone movements that were detected depending on the object type and the distance of the user from the plane of the antennas. In general, we find that the reliability of detection is high. The reliability is a little lower for the box carried in the hand as compared to the badge left hanging since the operation of the tag is interfered by direct contact with the hand.

### *7.1.2 Object Interaction Detection*

Next, we present an evaluation of object interaction detection in SixthSense. We placed two RFID antennas on a table such that they faced each other. We placed three tagged objects — an empty water bottle, an empty coffee mug, and a cardboard box — at various locations on the table, in between the two antennas. So a larger separation between the antennas also meant that the objects were typically at a greater distance from either or both antennas. The experiment involved the user picking up each object in turn and interacting with it as one normally would. Each object was interacted with 6-7 times, for a total of 20 interactions

[2] Note that deploying such rogue RFID readers would be far more challenging for an attacker than say deploying rogue WiFi APs because of the much more limited range of RFID.

| Distance between antennas | Detection Time |
|---|---|
| 1.5m | 2.39s |
| 2m | 3.4s |
| 2.5m | 5.03s |

**Table 3: Detection of inter-zone movement.**

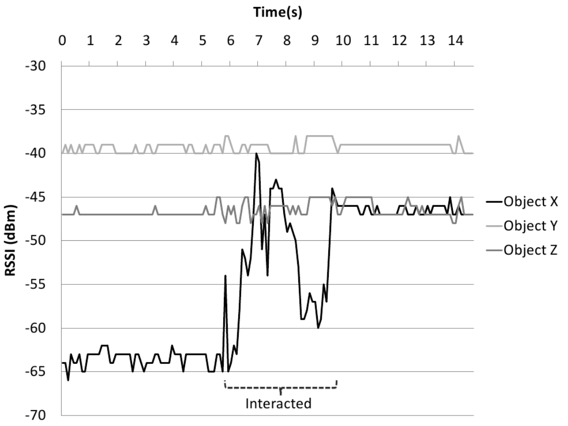


**Figure 8: Object Interaction - Antenna 1**

across all three objects. We recorded the time it took after the user started interacting with an object for the interaction to be detected. We also looked for false positives, i.e., spurious detection of interaction.

Figures 8 and 9 show the impact of interaction with one object (X) on the RSSI of all three objects as recorded at the two antennas. As expected, we see large swings in the RSSI of object X as recorded at both antennas. So measurements from either antenna would have allowed detection of this interaction. However, as Figure 9 shows, there is a significant swing (exceeding the 10 dBm threshold from Section 6.1.5) in the RSSI of object Z as well at around 5 seconds. Thus if we were to depend on antenna 2 alone, we would have detected a spurious interaction with object Z. However, since there is no corresponding swing in the RSSI of object Z as recorded at antenna 1, a conjunction of detections based on the two antennas would have helped avoid the false positive. This is the algorithm we use in our evaluation below.

Table 3 reports the detection time for different distances between the antennas. We find that detection takes longer, i.e., requires continued interaction of a longer duration, when the distance is larger. This is intuitive since the impact of the interaction on the RSSI would be weaker at greater distances. On a separate note, there were no false positives.

### 7.1.3 Testbed Deployment

Finally, we turn to the small-scale deployment in our testbed, which is described in Section 3.2. We deployed the RFID reader in such a way that each antenna covered one user's workspace. The deployment spanned four zones in all, each occupied by one user. We tagged the employee badges of the users as well as their objects such as cell phones, laptops, water bottles, books, and coffee mugs. We also installed the calendar, presence, and login monitors on the desktop machine of each user.

Besides spending time at their own workspace, users walked to the pantry or the conference room on occasion, in the process moving across multiple zones. The timing of these walks, which subset of their objects users carried with them, and when they returned to their workspace to log in were chosen artificially, to exercise various aspects of the SixthSense inference algorithms (e.g., on occassion, we had multiple users enter their workspaces and log in simultaneously).

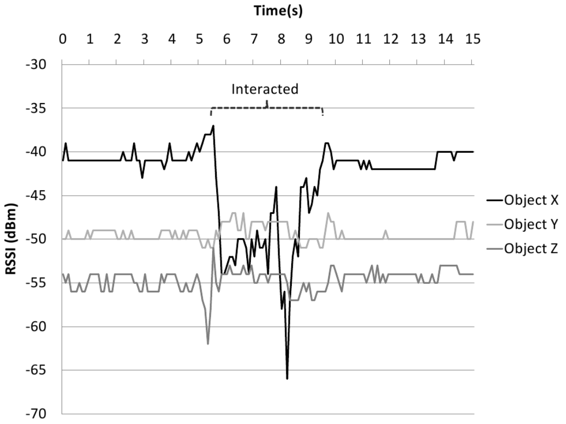


**Figure 9: Object Interaction - Antenna 2**

We collected several sets of traces of such activity and fed these to the SixthSense inferences algorithms. We found that person-object differentiation and ownership inferences were performed correctly. The average number of inter-zone movements of tags needed to make correct inferences was 4 and the average number of logins per user needed to establish the identities of all four users was 3.

## 7.2 Simulation-based Evaluation

Since we had only had a small-scale deployment, with a single reader, we evaluate the SixthSense inference algorithms at scale using the simulator mentioned in Section 3.3. The simulator uses a probablistic model to generate artificial traces of movement of tagged people and objects. While it is able to simulate inter-zone movement, the simulator does not model physical aspects such as the RSSI.

### 7.2.1 Person-Object Differentiation and Object Ownership Inference

In this experiment, we simulate 20 users, each owning 4 objects, so that there are a total of 100 tags in the system. We assume that each user is has a home zone of their own and that the zones are laid out in a line (akin to rooms along a hallway) so that moving from zone $i$ to zone $j$, where $i < j$ and $i, j \in [1, 20]$, would involve passing through all zones $k \in (i, j)$.

We simulate the movement of users as follows. At each timestep, a user who is in their home would decide whether to start walking with a probability of 10%. If the decision is to walk, a destination is picked uniformly at random according to the desired average walk length. Then in each timestep, the user advances towards the chosen destination by one hop. After reaching the destination, the user turns around and heads back to their home zone, again one step at a time. During each walk, the user carries along 0 to 4 of their objects, picked at random. We assume that the RFID infrastructure is able to reliably detect and record inter-zone transitions made by all tags.

Figure 10 shows how well person-object differentiation and object ownership inference perform as the average count of

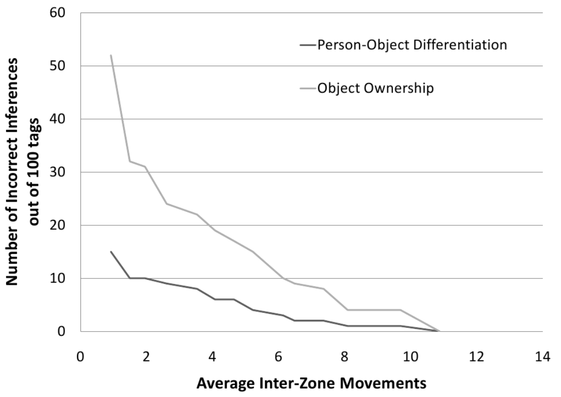


**Figure 10: Simulating people movement with an average walk length of 10**

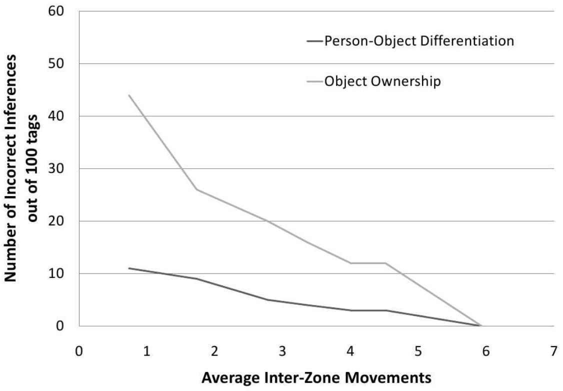


**Figure 11: Simulating people movement with a walk length of 2**

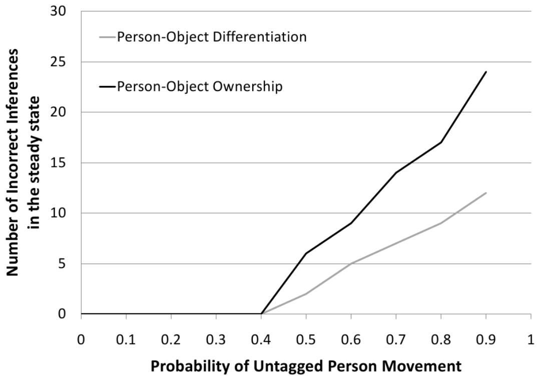


**Figure 12: Effect of untagged people movement**

inter-zone movement of tags is varied. The average walk length of a person in this experiment was 10 inter-zone transitions for the round-trip from their home zone to the destination, and then back to their home zone.

We then repeat the same experiement, with the walk length of each user set to 2 instead of an average of 10 (i.e., a user's walk involves going to a neighbouring zone and then back). As shown in Figure 11, our algorithm is able to make correct inferences with fewer average inter-zone movements compared to the case in Figure 10. Note that a user carries the same set of objects throughout a walk but can change the set of objects from one walk to the next. So for the same total number of inter-zone movements, a larger number of shorter walks results in a greater diversity in the combinations of people and objects that move, providing a richer set of observations that enable the SixthSense inference algorithm to perform better.

We also evaluate the SixthSense inference algorithms in anomalous situations. A user may occasionally leave their badge in their workspace, and walk down to a colleague's office or to get a cup of coffee. In such a situation, the tagged objects carried by the untagged user would seem to move by themselves. To evaluate the robustness of the inference algorithms in SixthSense to such misleading observations, we simulate the movement of untagged people and vary the probability of its occurrence. As shown in Figure 12, our algorithms continue to make correct inferences as long as the probability of a user moving when untagged is under 0.4.

### 7.2.2 *Person Identification*

Next, we turn to evaluating the person identification algorithm. We simulate a population of users, each with a home zone that has already been identified. We simulate 10% of users entering their workspaces simultaneously at any point in time. Note that 10% is a large fraction in this context and is a conservative assumption since multiple users entering their workspaces simultaneously creates ambiguity when inferring the identities of the individual users. We vary the probability with which users log in upon entering their workspace. We evaluate our person identification algorithm with 1000, 100 and 10 people, as shown in Figure 13. In a setting with 100 users, each of whom logs in with a probability of 0.5 (i.e., 50% of the time) soon after entering their workspace, the average number of logins needed per user for SixthSense to infer the identity of all users is 17. If we reduce the fraction of users who enter their workspaces simultaneously to 5%, the average number of logins needed per user drops to 10.

## 8. PROGRAMMING MODEL

Many applications can be built by leveraging the SixthSense inferences. In this section, we briefly explain the programming model that SixthSense provides for such applications to be built. SixthSense exposes a set of events (callbacks), which an application can subscribe to, and a set of lookup functions, which an application can call to get information from the processed database. We believe that this basic set of events and lookups are sufficient to build a rich set of applications.

Note that the SixthSense APIs refer to a tag by its tagID, which corresponds to the unique EPC for each tag in our prototype system.

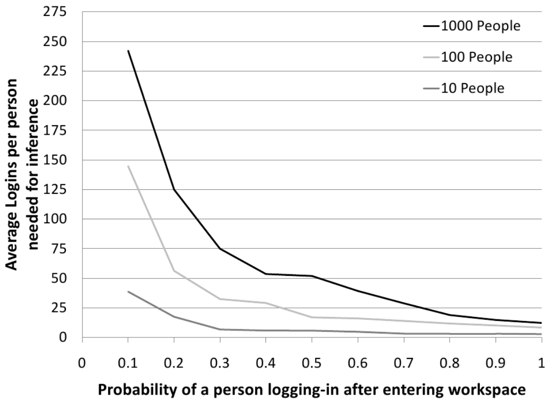


**Figure 13: Person identification**

## 8.1 Events

The system provides the following events to which applications can subscribe:

- `InterZoneMovementEvent(tagID, startZone, endZone, Time)`: This event is raised when a tag moves across zones.
- `ObjectInteractedEvent(tagID, Zone, Time)`: This event is raised when someone interacts with a tagged object.

## 8.2 APIs

SixthSense exposes the following set of APIs:

- `GetTagList()`: Returns all the tagIDs in the system
- `GetPersonTags()`: Returns all the person tagIDs in the system
- `GetOwnedObjects(tagID)`: Returns all the object tagIDs owned by a person
- `GetTagType(tagID)`: Returns the type of a tag, i.e., person or object
- `GetTagOwner(tagID)`: Returns the owner of the specified object tag
- `GetPersonTagIdentity(tagID)`: Returns the user name corresponding to the specified person tag
- `GetZoneType(Zone)`: Returns the type of a zone (private/shared/reserved/public)
- `GetTagsInZone(Zone)`: Returns the list of tags currently present in a zone
- `GetTagWorkSpaceZone(tagID)`: Returns the set of zones identified as the workspace of a person
- `GetCurrentTagZone(tagID)`: Returns the zone that a tag is currently in. Returns unknown if the tag is currently not found.
- `GetCalendarEntry(ID, Time)`: ID can be a person ID or a conference room ID. It returns appointment information for a give time in a person's or conference room's calendar.
- `SetCalendarEntry(ID, StartTime, EndTime, Location)`: This API is used to automatically add an appointment in the calendar corresponding to the ID (user or conference room) for the specified time and location.

# 9. APPLICATIONS

In this section, we detail a few applications that we have built on top of the SixthSense system. These applications use the APIs and events provided by the SixthSense programming model described in Section 8.

## 9.1 Misplaced Object Alert

The misplaced object alert service tracks the movement of users and objects, and notifies a user when it thinks that he/she may have misplaced an object. An object is said to be misplaced when it is in a shared workspace or a public area, and the object's owner, who was also in the same zone, moves away to a different zone. When such a situation is detected, the system suspects that the object may have been misplaced and alerts the user via email or phone.

To implement this service, we first invoke `GetTagList()` to learn of all the tags in the system. We then invoke `GetTagType()` on each tag to learn whether it corresponds to a person or an object. For the object tags alone, we invoke `GetTagOwner()` to learn the tagID of its owner. Finally, we register for the inter-zone movement event on all person tags. Armed with the ownership and movement information, we are in a position to detect whether an object is misplaced, as noted above. If it is, we invoke `GetPersonTagIdentity()` on the affected person tagID to learn the identity of the corresponding person and then alert him/her.

We can define simple variants of this service that, say, insist on a minimum separation in terms of time or distance between an object and its owner, for the object to be deemed as having been misplaced. Using the same set of events and APIs as above, a user can also query the system for information about the objects he/she owns. For example, the user can ask the system: "When was the object I had with me around 4:30 PM last detected?"

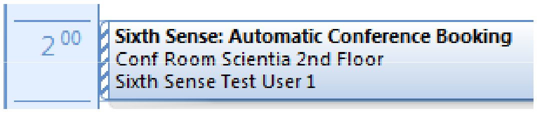
## 9.2 Annotated Video

We have built a simple application that the annotates the video feed from a camera with RFID events corresponding to inter-zone movement and object interaction. To annotate a video with events, we store the video as an AVI file. Separately, we record an event log that is temporally correlated with the video but is separate from it. Besides the time of occurence, an event also records the tagID involved and the zone where the event took place. To support rich querying, we could also store additional information regarding each event obtained from invocations of `GetTagType()`, `GetTagOwner()`, and `GetPersonTagIdentity()`, as appropriate. When a query is issued for a specific event, we extract the corresponding time offset from the event log and seek directly to the corresponding point in the video.

One application of such an annotated video solution is annotating the video recorded by the security cameras in a building. The authors have had the occassion in the past to sift through hours of security video recordings for want of a suitable indexing mechanism. With our annotated video application, a user who is searching for an untagged object that he/she has misplaced could, for instance, query for all sections of the video recording corresponding to when he/she was in the zone of interest.

## 9.3 Automatic Conference Room Booking

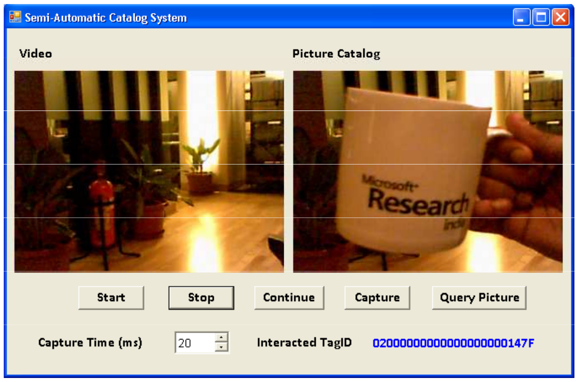
SixthSense automatically identifies a shared space as a reserved space (e.g., a conference room) using user calendar



2 00

Sixth Sense: Automatic Conference Booking
Conf Room Scientia 2nd Floor
Sixth Sense Test User 1

**Figure 14: Automatic Conference Booking**



Capture Time (ms) 20

Interacted TagID 02000000000000000000147F

**Figure 15: Semi-Automated Image Catalog**

information, as described in Section 6.1.3. While reserving such a room may be the norm, people may sometimes occupy it without reservation, if it is not already occupied and they are looking for space to hold an unplanned, last-minute meeting. Despite the barging in, if the room has in fact not been reserved for the time in question, it would be desirable to reserve it for the group that has barged in, to avoid the possibility of someone else trying to reserve the apparently free but in fact occupied room.

We have built an automatic conference room booking application to accomplish precisely this. From SixthSense's zone identification procedure (Section 6.1.3), we first recognize that the room in question is a reserved space. We then invoke `GetTagsInZone()` to detect all the tags in the current zone. Invocations of `GetTagType()` on each such tag would identify the people, if any, present in the room. If we find a group of people staying in the room for longer than say 5 minutes, we presume that they are meeting in the room, and so invoke `GetCalendarEntry()` on the room to see if there is a reservation for the current time. (Note that it is common practice to treat reserved rooms also as persons, with their own calendars.) If there is not one, we create an automatic reservation by invoking `SetCalendarEntry()` for the room and each person in the meeting. Figure 14 shows an example of such an automatic reservation created by our application.

## 9.4 Semi-Automated Image Catalog

We have also built a semi-automated image cataloging application that works as follows. The user picks up a tagged object, holds it in front of their camera, and clicks a picture. By looking for `ObjectInteractedEvent()` events in the zone that the user is currently in, we can automatically identify the tagID of the object that was picked up. (If SixthSense finds that more than one object was interacted with, the user could be alerted to repeat the process, if they so desire.) This tagID is then cataloged along with the picture that was just taken. Figure 15 shows a screenshot of our semi-automated image cataloging system. A catalog such as this would allow users to identify their objects more naturally using images rather than tagIDs.

# 10. DISCUSSION

In this section, we consider various issues pertaining to achieving widespread RFID coverage in the workplace, as assumed by SixthSense.

## 10.1 Economic Feasibility

With the growth in the market opportunity in RFID, and the maturing and standardization of RFID technologies, the prices of both RFID readers and tags have been declining. The price of a passive RFID tag is expected to drop to 5 U.S. cents in volume production [9]. On the reader front, Intel recently announced [5] the integration of a substantial number of RFID components into a single chip (Intel's R1000), leading to the expectation that the price of UHF RFID readers would drop to just U.S. $500 from about U.S. $1600 today. (Note that in a typical configuration, a single RFID reader would have 4 antennas attached to it with cables long enough to allow the antennas to cover non-overlapping spaces.) We can expect prices to drop further if past trends for other technologies such as WiFi is any guide. In fact, compared to WiFi, the "client end" in RFID, i.e., the tag, is very inexpensive, so it is only a question of the one-time cost for the infrastructure becoming affordable.

Even a limited deployment of RFID, covering say just high traffic areas such as conference rooms and stairwells, could provide some of the benefits of a system like SixthSense, although some of the automatic inferencing may have to be replaced with manual input. In fact, RFID readers in such spaces could subsume the role of other sensors such as motion detectors that are often deployed in such spaces.

## 10.2 Privacy Implications

The unique identifier (e.g., EPC) carried by RFID tags coupled with the non-line-of-sight operation of RFID implies the ability to "see" and track objects far beyond what the human eye can do. For instance, an RFID reader could detect and identify objects that are inside a bag (termed *X-Ray vision* in [25]), with privacy implications as well as security implications (e.g., a thief can tell exactly which bags contain valuable items). Worse, a person and their possessions could be scanned repeatedly without them realizing it because of non-line-of-sight operation.

Juels [19] provides a survey of recent research on practical approaches to address the privacy problem, which we summarize here:

- *Killing tags:* As noted in Section 2, it is possible to kill a tag, thereby rendering it unreadable, by issuing a special reader command along with a password. While this might be suitable in some settings (e.g., consumers getting the tags on their purchases killed), this would not be applicable in settings where the ability to track objects in normal course is desired.

- *Renaming Approach:* Rather than being killed, tags could be renamed, perhaps repeatedly, to defeat attempts to track them surreptitiously. One possibility is for the reader to *relabel* tags. An alternative approach is for a tag to maintain multiple *pseudonyms* known only to legitimate readers and to cycle through them when queried. Of course, this would require more advanced tags.

- *Proxying Approach:* Users could carry their own privacy-enforcing *guardian* devices for RFID, say integrated with their mobile phones, that act like personal RFID firewalls and intermediate reader requests to their tags.

- *Distance Approach:* If a tag is able to estimate its distance from the reader, say using the signal-to-noise ratio of the reader's signal, it could estimate how far the reader is and reveal little when scanned from afar.

SixthSense could employ any approach that is practical. In our current prototype, we use a simple renaming scheme based on relabeling, as discussed in Section 6.4.

## 10.3 Health Implications

As with other forms of RF technology, RFID has evoked concerns about the health impact of exposure to a reader's RF transmissions. The primary concern with regard to RF fields in the 10 MHz to 10 GHz range (which includes the UHF band used by far-field RFID readers) is the heating of tissue due to the absorption of radiation. However, a scientific review by the World Health Organization (WHO) [2] concludes that the RF field limits for safety are well above the levels found in the living environment and that there is no convincing evidence that exposure to RF shortens the life span of humans, or induces or promotes cancer.

Specifically with regard to RFID, although the transmitted RF power (up to 2 W) could be higher than that from other sources such as WiFi access points (up to 1 W but often just 100 mW), it is still well within safe limits even when a subject is just a few centimeters from the antenna [6].

Thus, based on current knowledge and understanding of the impact of RF fields on human health, there is no cause for concern, although this question will undoubtedly continue to receive much attention and study.

# 11. RELATED WORK

There exist a large body of work on measurement of RFID performance. Ramakrishnan et al. [23] report on the read performance of different tags in terms of read range, read rate, orientation sensitivity, and environmental factors such as the impact of crowding (i.e., many tags in the vicinity) and the presence of water or metal. They report a relatively sharp drop off between a 100% response rate in the in-field region and a 0% response rate in the out-of-field region, with a relatively narrow weak in-field region sandwiched in between. As reported in Section 4, our experiments suggest a much sharper drop-off. They also report on the significant impact of tag orientation, which is consistent with our findings. However, while the experiments in [23] are inspired by the supply chain setting (e.g., testing on a conveyer), our experiments are set in an office building and include objects one might find in such a setting (e.g., laptops, mobile phones, etc.). Furthermore, our state-of-the-art reader enables measurement of RSSI, which is not reported in [23].

Rahmati et al. [22] also report on measurements of tag read reliability. They report a much more gradual drop-off in the read response rate (i.e., a much wider weak in-field region) compared to [23]. To improve reliability, they study the impact of attaching multiple tags to an object. They report that attaching multiple tags in different orientations and with an inter-tag spacing of at least 20-40 mm (to avoid interference), can improve read reliability by over 50%. As noted in Section 6.3, we use the same multi-tagging approach to improve read reliability in SixthSense, and also develop a simple algorithm to automatically identify the group of tags attached to the same object.

There has been much research interest in RFID-based localization because, unlike conventional systems like GPS, infrared sensors (e.g., Active Badge [14]), and wireless LAN RF (e.g., RADAR [12], RFID allows any and all objects, not just computing or communication devices, to be tagged very cheaply and thereby tracked. Hahnel et al. [18] propose a method to localize RFID tags using a mobile platform to automatically generate tag maps. These maps are subsequently used to localize robots people.

Ferret [20] is a system for finding locations of mostly static objects augmented with RFID tags, by interatively refining observations made from multiple locations and in different directions by a mobile RFID reader, which is assumed to know its own location. This setup is integrated with a camera, which allows the image to be annotated with the estimated locations of RFID tagged objects and displayed in real time. Our annotated video application (Section 9.2) is related to this but does not require a mobile reader, although it settles for zone-level localization rather than actually pinpointing objects within a video frame as in Ferret. Furthermore, it leverages SixthSense's ability to automatically infer the identities of and the relationship between tags based on their mobility pattern and information from other enterprise sensors and systems. So it can automatically answer a query like "show me all video frames where person A interacted with object X." In contrast, Ferret would need the actual identity of each tagged entity to be known a priori.

Several scenarios in ubiquitous computing require automatic inferencing of what a person is doing or intends to do. One of the key objectives of SixthSense is to provide such inferences in enterprise environments using RFID technology and other available sensors and systems. In the past, researchers have applied three main techniques to human-activity inference: computer vision, active sensor beacons, and passive RFID. While vision based inferencing techniques suffer from robustness and scalability problems, active sensor beacons require batteries. Approaches based on passive RFID tags avoid these difficulties, making them particularly attractive. Smith et al. [24] present pioneering work on RFID-based human-activity detection. While the proposed techniques can provide rich information about interactions, these are either obstrusive, or require non-standard, customized RFID tags or devices (e.g., a special glove or bracelet to be worn by the user). Assuming that reliable detection of people-object interactions is possible, Philipose et al. [21] present a framework for inferring *activities* from such interactions. For example, if it can be detected that a person interacted with tea, water, and sugar, can we infer that the person is trying to make tea? The solution proposed uses RFID coupled with data mining techniques and a probabilistic inference engine to recognize activities.

There has also been work on alternative, unobtrusive techniques for detecting interaction with RFID-tagged objects. Fishkin et al. [16] use variations in the response rate (i.e., the fraction of reads that are responded to) of individual tags and groups of tags to detect interaction. However, as reported in [23], we find that there is typically a sharp drop off in the response rate from 100% down to 0%, making it hard to use response rate for detection. Instead, as presented in Section 6.1.5, we use variations in the RSSI of a tag's response to infer interaction.

The deployment of RFID-based systems in public or enterprise environments raises several important issues and challenges, as noted by Welbourne et al. [28]. They discuss a range of deployment challenges having to do with the mounting of tags, the positioning of antennas, the use of multiple antennas for redundancy, compliance with health regulations, and privacy concerns.

In addition to the specific points of differentiation made above, the key distinction of SixthSense from prior work is that it combines RFID information with information from other enterprise sensors and systems to make inferences automatically.

## 12. CONCLUSION

We have presented SixthSense, an enterprise intelligence system that extends the reach of RFID to enterprise settings, where there is a rich interaction amongst people, objects, and workspaces. By combing RFID events with information obtained from other enterprise systems such as a shared calendar, SixthSense is able to make several inferences automatically. For example, it can automatically differentiate between person and object tags, and learn who the likely owner of an object is. Our limited evaluation of SixthSense in a small testbed and our more extensive evaluation in simulation confirm the efficacy of SixthSense's inference algorithms.

We have built SixthSense as a platform, exposing a rich set of API, on top of which a range of applications can be built. We have presented a few of these applications, including a misplaced object alert service and enhanced calendar.

In future work, our goal is to deploy SixthSense on a more extensive RFID setup that also covers shared spaces such as conference rooms and hallways. We also plan to refine our programming model, as we gain experience with more applications.

## Acknowledgments

Srikrishna R. helped with some of our initial RFID benchmarking measurements. Ranjita Bhagwan, Ram Ramjee, and Geoff Voelker provided useful feedback at various stages in this work. Our MobiSys shepherd, Gaetano Boriello, and the other anonymous reviewers provided insightful comments on an earlier draft of this paper. We thank them all.

## 13. REFERENCES

[1] EPCglobal. http://www.epcglobalinc.org.

[2] Health Effects of Radiofrequency Fields. Fact Sheet N183, World Health Organization, May 1998, http://www.who.int/docstore/peh-emf/publications/facts_press/efact/efs183.html.

[3] Impinj Monza Gen 2 Tag Silicon for UHF RFID. http://rfid-support.impinj.com/Common/FileTransfer/Download.aspx?ObjectId=30089219&env=807989.

[4] Impinj Speedway Reader. http://www.impinj.com/uploadedFiles/RFID/RFID_Products/Impinj-Speedway_Reliable_Data(singlepages-web).pdf.

[5] Intel's RFID Move To Slash Reader Prices. http://www.eweek.com/article2/0,1759,2100828,00.asp, Mar 2007.

[6] Is RFID Safe at the Work Place? RFID4SME Report (project funded in part by the European Union), http://www.rfid4sme.eu/.

[7] Microsoft BizTalk RFID. http://www.microsoft.com/biztalk/technologies/rfid/default.mspx.

[8] RFID 301: A Detailed Look At Using RFID In Your Library. 3M Corporation, http://solutions.3m.com/wps/portal/3M/en_US/library/home/resources/case_studies/rfid_301/.

[9] SmartCode Corp. Announces the World's First 5-cent RFID Tag. http://www.smartcodecorp.com/newsroom/01-05-06.asp, May 2006.

[10] The History of RFID Technology. RFID Journal, http://www.rfidjournal.com/article/view/1338/1/129.

[11] The Internet of Things. ITU Internet Reports 2005, http://www.itu.int/osg/spu/publications/internetofthings/.

[12] P. Bahl and V. N. Padmanabhan. RADAR: An Inbuilding RF-based User Location and Tracking System. In *INFOCOM*, 2000.

[13] N. Chaudhry, D. R. Thompson, and C. W. Thompson. RFID Technical Tutorial and Threat Modeling Version 1.0. Department of Computer Science and Engineering, University of Arkansas, http://www.csce.uark.edu/drt/publications/rfid-tutorial120608.pdf, 2005.

[14] R. W. et al. The active badge location system. *ACM Transactions on Information Systems*, Jan 1992.

[15] K. Finkenzeller. *RFID Handbook*. Wiley & Sons LTD, 2003.

[16] K. P. Fishkin, B. Jiang, M. Philipose, and S. Roy. I Sense a Disturbance in the Force: Unobtrusive Detection of Interactions with RFID-tagged Objects. In *Ubicomp*, 2004.

[17] M. Geist. Computer and E-Mail Workplace Surveillance in Canada: The Shift from Reasonable Expectation of Privacy to Reasonable Surveillance. Prepared for Canadian Judicial Council, http://www.cjc-cm.gc.ca/cmslib/general/Geist_report.en.pdf, 2002.

[18] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. Mapping and Localization with RFID Technology. In *IEEE ICRA*, 2004.

[19] A. Juels. RFID Security and Privacy: A Research Survey. *IEEE JSAC*, Feb 2006.

[20] X. Liu, M. D. Corner, and P. Shenoy. Ferret: RFID Localization for Pervasive Multimedia. In *Ubicomp*, 2006.

[21] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hähnel. Inferring Activities from Interactions with Objects. *IEEE Pervasive Computing*, Oct 2004.

[22] A. Rahmati, L. Zhong, M. Hiltunen, and R. Jana. Reliability Techniques for RFID-Based Object Tracking Applications. In *IEEE/IFIP DSN*, 2007.

[23] K. M. Ramakrishnan and D. D. Deavours. Performance Benchmarks for Passive UHF RFID Tags. In *13th GI/ITG Conf on Measurement, Modeling, and Evaluation of Computer and Communication Systems*, 2006.

[24] J. R. Smith, K. P. Fishkin, B. Jiang, A. Mamishev, M. Philipose, A. D. Rea, S. Roy, and K. Sundara-Rajan. RFID-based Techniques for Human-activity Detection. *CACM*, Sep 2005.

[25] F. Stajano. RFID is X-ray vision. Technical report, Aug 2005. University of Cambridge Computer Laboratory, Technical Report No. 645, http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-645.pdf.

[26] V. Stanford. Pervasive Computing Goes the Last Hundred Feet with RFID Systems. *IEEE Pervasive Computing*, Apr 2003.

[27] R. Want. An Introduction to RFID Technology. *IEEE Pervasive Computing*, Jan 2006.

[28] E. Welbourne, M. Balazinska, G. Borriello, and W. Brunette. Challenges for Pervasive RFID-based Infrastructures. In *PERTEC 2007 Workshop on Pervasive RFID/NFC Technology and Applications*, 2007.