MAX: Human-Centric Search of the Physical World

Kok-Kiong Yap, Vikram Srinivasan and Mehul Motani
Dept. of Electrical and Computer Engineering,
National University of Singapore
kokkiong.yap@nus.edu.sg, elevs@nus.edu.sg, motani@nus.edu.sg

ABSTRACT

MAX is a system that facilitates human-centric search of the physical world. It allows humans to search for and locate objects as and when they need it instead of organizing them *a priori*. It provides location information in a form natural to humans, i.e., with reference to identifiable landmarks (e.g., on the dining table) rather than precise coordinates. MAX was designed with the following objectives: (i) human-centric operation, (ii) privacy, and (iii) efficient search of any tagged object. In the system, all physical objects, from documents to clothing, can be tagged and people locate objects using an intuitive search interface. To make search efficient, MAX adopts a hierarchical architecture consisting of tags (bound to objects), sub-stations (bound to landmarks) and base-stations (bound to localities). Tags can be marked as either public or private, with private tags searchable only by the owner. MAX also provides for privacy of physical spaces. MAX requires minimal initial configuration, and is robust to reconfiguration of the physical space. To optimize system performance, we present a methodology to design energy and delay optimal query protocols for a variety of device choices. We have implemented MAX using Crossbow motes and conducted user trials in a 5m by 5m cluttered office. The user feedback was positive, demonstrating the feasibility of MAX for human-centric search. We contend that a MAX-like search system will enable sharing (e.g., books on a college campus) and trading (e.g., buying and selling used books) of physical resources, and will be the engine for a host of new applications.

Categories and Subject Descriptors: H.4.0 [Information Systems Applications]: General

General Terms: Design, Experimentation, Human Factors, Performance, Security

Keywords: Human-centric, Search, Physical World, Landmark based Localization

SenSys'05, November 2–4, 2005, San Diego, California, USA.

1. INTRODUCTION

It is inarguable that information and communication technology have revolutionized the way we live. One perspective, which is particularly relevant here, is that many of these technologies allow humans to be more "disorganized", meaning they need to spend less time and effort in planning and organization without compromising on efficiency. The following examples capture this notion:

- Wireless communications technology allows people to meet up spontaneously, without agreeing *a priori* on a precise location to meet.
- Credit cards and automatic teller machines have eliminated the need to plan ahead and carry adequate amounts of money, at home or on vacation.
- Search engines such as Google have become so efficient at finding the information you want that it is now not necessary to download and save data locally.
- Gmail allows email to be labelled and searched efficiently, eliminating the need to organize and file your emails.

We contend that the ability to search the physical world will allow humans to be more disorganized with their physical belongings. It will have far reaching implications and applications similar to search engines for the virtual world. We propose MAX[1], a system and architecture that allows physical objects to be searched for and located quickly and efficiently. We assume that all physical objects can be tagged with small devices which have limited processing and communication capabilities. Current technology trends, such as smart paints, smart textiles, smart dust and RFID tags, point to a not too distant future when this will be possible. To ensure privacy, physical objects can be tagged as being either private or public, with private objects searchable only by their owner and public objects searchable by anyone. Moreover, the privacy of physical spaces can also be protected by marking areas as public, private or off-limits, with public spaces fully searchable by anyone, private spaces coarsely searchable (explained later), and off-limits spaces accessible only to their owners.

The ability to search the physical world will allow us to move away from the paradigm of "everything has *its* place" to a paradigm of "everything has *a* place". As an example,

[1]MAX is short for Maxwell's Demon who attempts to create order from disorder, allegedly violating the second law of thermodynamics.

an audio CD need not be put back into its specific jewel case every time; instead a CD can be put into any case and one can search for a particular CD when one wishes to listen to it. The ability to locate objects as and when we please will allow us to spend less time and effort in organizing our physical spaces and to use our limited space more efficiently. The search capability combined with the ability to distinguish between private and public objects mimics how information is shared in the digital world. We predict that this will enable a variety of new applications which involve the sharing and trading of physical resources. For example, in a university campus scenario, expensive and hard to find books can be shared between professors and students. In addition, used text books can be merely tagged as being for sale, without having to go through the effort of advertising them.

In this paper, we describe MAX which is a system and architecture for searching the physical world. The key idea underlying MAX is that it is meant to be used by humans and thus it provides information in a form easily accessible to humans. What we mean by this is that when humans locate objects, they do not do so in terms of absolute co-ordinates, rather they use identifiable landmarks. For example, one says "my notebook is on my desk" or "the keys are on the dining room table". Before delving into the detailed system architecture, we note that MAX is the result of the following design goals:

- **Human-centric Operation:** The system should be simple to install and easy to use. Central to the theme of a user-friendly system is the use of natural human language, which allows a user to interact with the system at ease. Natural human language should be accepted as input and the results of the system should also be presented as intuitively as possible.

  Moreover, the system must be robust to reconfiguration of physical spaces and should require minimal or no changes when such reconfigurations occur. For example, a person might wish to move their furniture around or move out to a new apartment or office space.

- **Security and Privacy:** Security requires that the system is not vulnerable to unauthorized access. Privacy involves several considerations. First, the movement and locations of privately owned objects should not be continuously tracked. Second, objects labelled private should only be searchable by their owners, while public objects can be searched for by anyone. Finally, the system should provide for varying degrees of access to physical spaces by allowing spaces to be marked as off-limits, private, or public.

- **Efficient Search:** The system must allow thousands of objects to be searched across wide areas (e.g., office buildings, university campuses or even across a city). Search results must be returned quickly and the search process should utilize the system resources such as energy, bandwidth and memory as efficiently as possible.

There have been several proposals for localization of wireless tags. These proposals can be either classified as *location tracking systems* such as [1–3] or *location support systems* such as [4]. Location tracking systems are proposed for applications such as inventory tracking. In these proposals, the system proactively keeps track of the location of the objects and stores them in a central database. The fact that object locations are stored raises concerns about privacy and also implies that it will be difficult to scale. Location support systems [4] allow users equipped with wireless devices to locate services in their vicinity (e.g., a printer). In addition in location support systems, wireless tags estimate their locations themselves and these locations are not stored in a central database. Therefore, these systems do provide some level of privacy. However neither of these systems satisfy our design goals. They do not provide a mechanism to distinguish between private and public objects or private and public physical spaces. In addition, they are not robust to reconfigurations of the physical space. In order to provide accurate location information, these systems require carefully placed beacons and a context aware middleware layer with map servers to provide human-centric information.

The architecture we propose in this paper is based on two observations. First, technology trends point to a not too distant future when small tags with limited processing and communications capabilities will be embedded everywhere. Second, humans are powerful sensors who require only approximate coordinates in the form of cues and landmarks to locate objects. These two simple observations allows us to design an architecture that satisfies all our design objectives. We assume that all objects can be tagged. In each tag is stored a label (or descriptor), which describes the object to which it is attached. For example, a label could be "brown coffee table" or "blue denim shirt". We propose a hierarchical architecture, where the objects in each level may be more mobile than the objects in the higher tiers. For example, the tag associated with a room can be at the top of the hierarchy, while the furniture in the room at the next level down and all other objects within the room at the lowest level of the hierarchy. Whenever a search is made (e.g., cellular phone), it is localized with respect to objects in the higher levels of the hierarchy (e.g., coffee table, living room).

In this section, we have motivated MAX as a system that facilitates human-centric search of the physical world and is based on several clearly identified design goals. In the rest of this paper, we present the design, analysis, and implementation of MAX. Our specific contributions in this paper are:

1. Proposal of a simple hierarchical architecture consisting of base-stations, sub-stations and tags.
2. Introduction of a new notion of privacy of physical spaces and providing for privacy and security of both physical spaces and objects.
3. Development of a simple prototype based on the Crossbow motes platform and conducting user trials.
4. Design of energy and delay optimal query protocols for different device choices.

Contributions 1, 2, and 3 can be found in sections 2, 3, and 4 respectively. Contribution 4 is included in sections 5 and 6. We end with related work in 7 and reflect deeply on the system we have designed in 8.

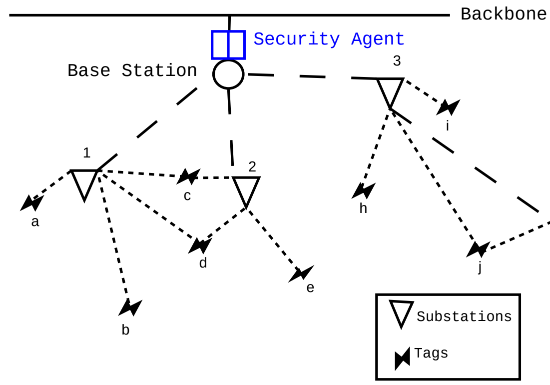


Figure 1: The architecture of the system depicting the five elements of the architecture.

2. SYSTEM ARCHITECTURE

We have argued that humans are powerful sensors who are able to locate objects quickly based on cues and identifiable landmarks. The advantage of bringing the human into the picture is that it considerably reduces complexity, resulting in a simple and scalable architecture. In our architecture, we assume that all physical objects can have a wireless tag attached to them and that each tag stores a description of the physical object to which it is attached. These objects can be located over a large geographical area, which we assume is divided into basic units (e.g, a room or office cubicle) called *localities*.

The architecture of MAX within a locality, shown in Fig. 1, is hierarchical in nature with different logical types of tags at each level. This three-tier hierarchy consists of base stations which are tied to a locality; sub-stations which are tied to mainly static objects (e.g., chairs, tables, shelves); and tags which are tied to small mobile objects (e.g., keys, books, phone, documents). The naming of these entities is a one-time process initiated by the owner and requires no further maintenance by the owner. If the object changes ownership, the new owner can rename the object.

The three-tier hierarchy is motivated by the way that humans organize and describe the locations of their things. Typically, people first localize an object as being in their room or office, and then describe where it is relative to easily identifiable landmarks such as their bed or desk. In terms of ease of use for the consumer, having less device types leads to more flexibility and device reuse. With this in mind, the four basic elements to this architecture are discussed below.

1. *Base-Station:* The base-station is at the highest level of the hierarchy in the system architecture. Logically, the base-station represents a locality which is immovable, such as a room, and it stores a descriptor of this locality (e.g., Jack's office). Depending on the size of the locality (e.g., a room), there could be one or more base-stations per locality. In addition the base-station also acts as a gateway between the backbone network and the wireless tags. Note that logically, the base-station describes objects which are static and immovable.

2. *Sub-station:* A sub-station is at the next level of the hierarchy and logically describes objects which are largely static and change positions occasionally (e.g., furniture). Sub-stations also store the label or descriptor of the object they are attached to (e.g., coffee table).

3. *Tags:* Tags are at the lowest level of the hierarchy. Logically these devices are attached to objects which are easily movable (e.g., books). Each tag stores a descriptor of the object it is attached to (e.g., Book Harry Potter). Multiple descriptor words are allowed in each tag, enabling users to label the object sensibly so that others can locate the object based on the label. It is the responsibility of the owner to label sensibly if she wants the object to be found in a search.

4. *Security Agent:* Security agents are software agents which reside at the base-stations, as illustrated in Fig. 1. Users are authenticated before they are allowed to query the locality served by the base-station. Handshaking based on public key cryptography is used to prevent any unauthorized access of the locality, allowing the owner of the locality (and base-station) to grant and deny access as needed. Thus localities can be made off-limits to unauthorized users, giving us the ability to protect the privacy of physical space. Security agents also allow the locality owner to mark an area as private, in which only coarse localization (e.g., whether the object is in the locality or not) is provided.

Query terminals: The above architecture allows users to query the physical environment. The interface to do so is provided by *query terminals*. Here, the results of the search must be presented to the users in the most natural way possible. The human-computer interface should also be intuitive to most, if not all. The interface is platform independent, allowing devices such as desktop computers, personal digital assistants, and mobile phones to be used as query terminals.

Object privacy: While the security agents described above provide user authentication and area privacy, *object privacy* is provided for in the system by allowing tags to be marked as private or public. Private tags are searchable only by their owner, while public tags are accessible to any user.

Wide area and remote search: So far, we have described search within a single locality, such as a room or office. However, MAX is able to support wide area search, meaning that a user can search for objects which are located in different localities, and remote search, meaning that a user can place a query from an arbitrary location. We envision that a network backbone connects the various base-stations and query terminals together. This network backbone could easily exploit existing network connectivity, such as LANs, DSL, and WiFi. This allows for the scaling of the system across a wide geographical area.

Writers: Programming descriptors into the tags and substations is a simple process accomplished using a writer, which can be integrated with the query terminal. We envision two ways in which a physical object can be tagged. The most likely way is for the descriptors to be programmed by the user at deployment. Alternatively, with the cooperation of manufacturers (e.g., furniture makers), objects can be embedded with a tag and description at the time of manufacture.

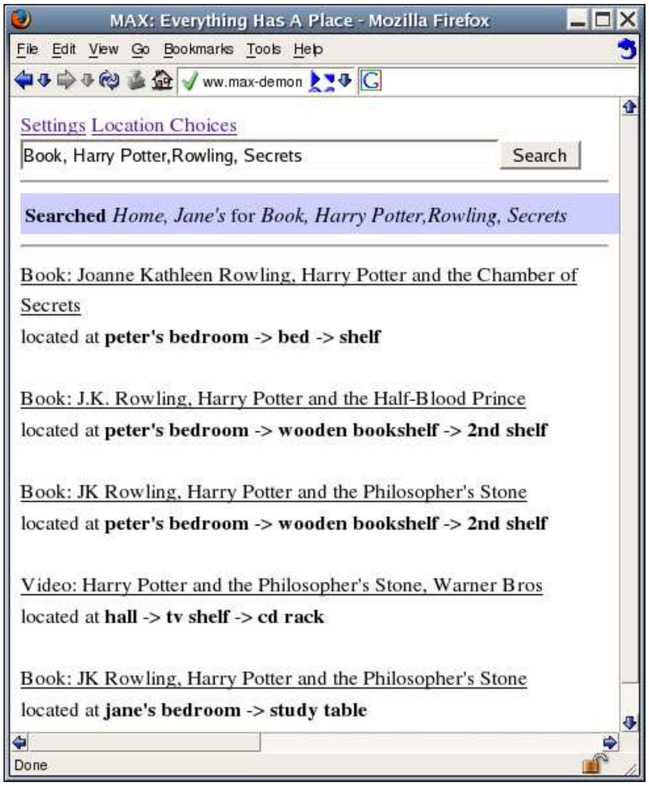


Figure 2: System Interface and Search Result Example

2.1 System Operation

In this section, we give a high level overview of how the system will be deployed and function. The exact details will be discussed later.

When a user wishes to find an object, he/she places the query via a simple interface on the query terminal. In Fig. 2 for example, the query could be "*Book, Harry Potter, Rowling*". This query is then propagated to one or more base-stations, who then broadcast the query to the sub-stations and tags in the locality. This broadcast mechanism could be over multiple hops. For example, the base-station could first transmit the query to the sub-stations who in turn transmit it to the tags. The tags whose descriptors match one or more of the query words respond to the base-station. The sub-stations listen in on the tag responses, compute the received signal strength indicator (RSSI) of these responses, and report these RSSI values to the base-station. The user then sees, at the query terminal, the results of the tags which have the maximum matches (for example, there could be many objects which have the label book and there could be Harry Potter DVD's, etc.) along with their estimated relative location. The relative location is estimated by associating a tag with the sub-station which hears it with the maximum RSSI value. A sample response is shown in Fig. 2.

Please note that in the above discussion we have only given a high level overview and not the precise mechanisms of the query protocol etc. Given the notorious nature of the wireless channel, one concern may be using RSSI values for estimating location. As we discuss in sections 4.3 and 8, RSSI is indeed sufficient for coarse localization.

2.2 Performance Metrics and Terminology

We consider three basic metrics which are

(i) *Latency* - the time elapsed from the entry of the query statement to when the results are returned.

(ii) *Energy* - the average amount of energy consumed by the battery-powered devices for a query.

(iii) *Match count* - the number of descriptors on a tag which match some part of the query.

The usefulness of the first two metrics is self-evident. The last metric is useful from a user perspective to determine which search result most likely corresponds to the object the user has queried for.

We discuss below some of the other terminology used in this paper.

- *Relevance:* Ideally, the results returned to the user must be of *maximal relevance* to the user. By maximal relevance, we mean that the sum of the match counts across all results returned to the user is maximized. Alternatively, some *heuristic* can be used to determine the set of results, without any guarantee on maximizing the sum of match counts. For example in a heuristic method, each sub-station could independently decide to return only results which are above some minimum match count threshold. This can significantly reduce the overhead incurred by the system. In this paper, depending on how the results are returned, we classify results as being either *maximally relevant* or *heuristic*.

- *Overlap:* Overlap refers to the fact that a tag might be within transmission range of several sub-stations. A large overlap implies that each tag can be overheard by several sub-stations. Given the uncertainties in the wireless channel, it is clear that if there are large overlaps, then, due to the variations in the wireless channel, it is likely that a tag might not be associated with the nearest sub-station.

2.3 Device Choices

1. *Tags:* We expect tags to be extremely cheap and small. They should have either very limited or no constant power sources. As a result these devices will have extremely limited communication and processing capabilities. Given these constraints, the most obvious choice of technology is Radio Frequency Identification (RFID). RFID tags can be either active or passive. Passive tags are very simple devices which can only store information and require no power from batteries. Whenever an RFID reader radiates them with energy, they respond by transmitting whatever is stored in them. The advantage of such a device is that it requires no power supply and can last for very long durations. However, using such simple devices might not result in a scalable protocol. The reason is that, irrespective of what the query is, every tag will respond with its descriptors. In this paper, we explore an alternative where, passive tags have comparators which will allow them to compare the query with its descriptor and respond conditionally to a query. In ISO15693 [5],

the passive RFID tags are able to compare their identities with a transmission, indicating that such modifications are possible without adding external energy sources such as batteries. We explore the trade-offs in terms of energy consumption and delay later in this paper.

Depending on the power of the RFID reader and the frequency used, readers can be classified as proximity readers with a range of 5 inches, medium-range readers with a range of 15 inches and long range readers with a range of up to 10 feet. Since the range is limited, base-stations might not be able to hear responses from the tags. Therefore the sub-stations will have to be responsible for relaying information from the tags to the base-stations.

2. *Sub-Stations:* We envision using sub-stations which are powered by batteries and have limited processing and communication capabilities. Given that we have chosen customized passive RFID tags (customized in the sense that they have comparators which allow them to conditionally respond to queries) and that these tags have a maximum range of 10 feet, the sub-stations have two functions. First they must act as RFID readers to query tags in their vicinity. Second, they must relay the results of these queries back to a base-station. In addition they must be able to do some limited processing to filter out the results of the queries. In order to conserve energy they should also have the ability to route query results to the base-station over multiple hops using other sub-stations. Such devices are already available in the market. Skye-Tek [6] is one such device which integrates a Crossbow Mote with an RFID reader. One design issue is that of the choice of the range of the RFID reader. If we choose a large range, then we will require fewer sub-stations for a given area. This will also imply that each tag will be in range of many sub-stations. However, since we are using only RSSI values to indicate proximity, we are very likely to choose the nearest sub-station incorrectly. Therefore, for increased accuracy, we require (i) RFID readers with small range and (ii) for each tag to be in the range of very few sub-stations. On the other hand, for scalability and cost, we require as few sub-stations as possible. Our experiments and experience indicates that we require a reader which has a range of at most one meter.

3. *Base-Station:* The base-stations can obtain query responses from the tags via the sub-stations and/or can read the tags directly, using built-in powerful RFID readers. The base-stations will have significant memory and processing ability. Moreover since they are likely to be line powered, we envision that they can significantly reduce the processing and energy burden on the sub-stations and tags.

2.4 System Choices

A fundamental question we need to answer is how much of the computational, processing and storage burden we would like to push down to the base-stations, sub-stations and readers. It is intuitive that the more burden we push onto the lower levels of the hierarchy, the lower the delays will be. However, its effect on energy consumption has to be explored. The resulting requirements on the various devices would also translate into cost, which should be carefully considered.

At the same time, the density of sub-stations and the degree of overlap has to be investigated. These are closely linked through the range of the reader. Requiring higher density of the sub-stations would directly increase the cost of the deployment. However, a sparse deployment may result in high latency, large energy consumption and non-useful localization.

The choice of modality is another system choice that has to be made. The common modalities are ultra-wideband signals, radio-frequency (RF) signals and ultrasound. Each of these modalities has its strengths and weaknesses. We have made a choice of using RF with RSSI, which is discussed in section 8.

3. SECURITY AND PRIVACY

So far, we have argued that MAX is a system which facilitates human-centric search of the physical world. Such a system enables new applications based on sharing (e.g., books on a college campus) and trading (e.g., buying and selling used books) of physical resources. Consider a simple application of sharing books amongst your colleagues on a college campus. Now you might designate a certain set of your books as public, meaning that users are able to determine the existence of and location of these books. The rest of your books are designated as private, meaning that no one else besides yourself is able to locate these books. The usefulness and power of such an application is clear. The critical components in such new applications are security and privacy.

The primary security and privacy risk facing MAX is the fact that it uses wireless technology to communicate. Though there are subtle differences between security and privacy, we will use the terms interchangeably (when there is no risk of confusion).

We note that in MAX there are two connotations to privacy. The first is with respect to objects that one owns, meaning that we wish private objects to be searchable only by the owner. The second is with respect to the space that a user owns or uses. What we mean by this is that a user might view the ability of third party to search within a user's personal space as an invasion of privacy. We need to provide for secure authentication and access to guard the privacy of both physical spaces and physical objects.

We now discuss mechanisms to ensure both classes of privacy, namely privacy of space and privacy of objects. The former is provided by security agents and the latter by cryptographic techniques.

3.1 Privacy of Physical Objects

First, we design for the notion of object privacy in which an object marked private can only be searched for by its owner. Objects marked as public are searchable by anyone. This notion of public and private marking is as previously described. Each user is authenticated at the query terminal and a pair of public and private cryptographic keys are retrieved (or created for new users). The mechanism to ensure user integrity will be described in Section 3.4.

Given the hardware constraints of the tags and sub-stations, it is not feasible for any complicated authentication to be performed at these devices themselves. We thus propose that the query terminal/base-station do all the necessary encryption and decryption.

Private objects have their descriptors encrypted using the owner's public key while public objects have their descriptors stored as clear text. When an owner queries for her objects, the query statement is encrypted by the base station (mechanism to be discussed later) using her public key. The encrypted query and its plain text are then concatenated and sent to the sub-stations. The encrypted component matches the private objects of interest while the plain text component will match all the public objects of interest. All results are then returned to the query terminal, where the encrypted descriptors are decrypted with the owner's private key.

3.2 Privacy of Physical Space

In MAX, physical space can be designated public, private or *off-limits*. Any user can query for physical objects in a public space. At the other extreme, only an authorized user can search for objects in a space that is designated *off-limits*. The authentication is done by the security agents residing at the base-station as illustrated in Fig. 1. These agents will communicate with the query terminals via secure tunnels, using standard public key cryptography and key distribution algorithms such as Pretty Good Privacy (PGP). Users will be authenticated before they are allowed to query the locality served by the base-station. Public-key cryptography handshaking can be used to prevent any unauthorized "access" of the space. The space is thus off-limits to anyone but the owner of the space.

We contend that these two extreme notions of security for physical spaces (off-limits and public) are insufficient. To see this consider what happens when a user (say Bob) inadvertently leaves behind his object in another person's (say Alice) off-limits space. Then, in effect this object is not searchable by both Bob and Alice, meaning it is lost to Bob. In another scenario assume that Alice wishes to put up her laptop for sale. In order for interested buyers to locate Alice's laptop, Alice will have to mark her space public. Note however that these searches will do landmark based localization and return the location of Alice's laptop with respect to other objects in her house (e.g. Hi-Fi stereo). At the same time, Alice does not want to leave her laptop in a public space, which is physically insecure. In response, we introduce the notion of a private space. With this notion, when a person searches either for his private object or a public object in another person's private space, the searcher can only ascertain the presence or the absence of the object in the space and not its location. However if the owner of a private space searches for his private or public objects, the search results are localized with respect to landmarks.

It should be noted that a binary answer is sufficient for an item to be found or put up for sale. We believe that the notion of private space opens up many new possibilities for MAX.

If a space is marked private, then there are three classes of objects to consider, (i) public objects, (ii) private objects owned by the owner of the private space and (iii) private objects owned by a third party (e.g., a user might leave her cellular phone in another user's home/office). We illustrate through a series of examples how search results are returned to different kinds of users in a private space.

Table 1: Results returned for Each User during Search in Alice's Private Space

Querying User	Objects: Alice's Public	Alice's Private	Bob's Private
Alice	Landmark	Landmark	None
Bob	Binary	None	Binary
Charlie	Binary	None	None

3.3 Detailed Example for Private Spaces

The results returned for a public and *off-limits* spaces are relatively straightforward. A query to a public space always return complete results, while a query to an *off-limits* space will not return any result unless the user is an authorized user authenticated by the security agent.

There are three characters in this example, Alice, Bob and Charlie. Alice has designated her home as a private space. This space has some public and some private objects which Alice owns. In addition Bob has mistakenly left his cell phone (marked private) in Alice's home.

The security agent residing at the base-station in Alice's home authenticates and allows Alice to search for all public and private objects which she owns and the locations of these objects are returned to Alice. When Charlie does a search for public objects within Alice's home, the query is sent as a concatenation of the clear text query and the query encrypted with Charlie's public key. Since Alice's private objects are encrypted with her public key, only public objects in her home match the clear text query. In addition, the security agent at the base-station ensures that only a binary answer (presence or absence of the object) in Alice's home is returned to Charlie without landmark based location information (e.g., near Hi-Fi stereo system). The final case is that of Bob searching for his private object (cell phone) in Alice's private space. Once again, the query is sent as a concatenation of clear text and encrypted text (with Bob's public key). In this case, Bob's cell phone will match the encrypted query. However, once again only a binary result (presence or absence) of Bob's cell phone in Alice's home is returned to Bob. These various cases are summarized in Table 1.

3.4 Privacy Mechanisms

Having described the concepts and mechanisms for privacy of objects and privacy of spaces in MAX separately, we now collate these mechanisms in a single description for clarity. For this, we will outline the various privacy-related operations performed during a single query.

To place a query, a user will encrypt her query using her private key and distribute it to the desired base stations. Upon reception, each base-station will verify her identity by decrypting the query using her public key. If the space is *off-limits* and the user is deemed to be unauthorized, the query will be ignored. Otherwise, the query will be performed in the locality of that base station.

Before querying the locality, the plain text query is encrypted by the user's public key to allow querying of the user's private objects. Objects are marked as public or private using a single bit in the tag. The public objects will be queried with the plain text, while the private objects will be queried with the encrypted version. Thus, all public objects and the user's private objects will be matched and returned.

If the space is public, the base station will deliver complete results to the user at the query terminal, else binary results are returned.

3.5 Price of Privacy

It is fortunate that the price of privacy is relatively low. The security agent uses standard software authentication at the base-station, which is fully customized, line powered and connected to the high-speed backbone. Thus, there is no issue of lifetime, processing power, or communication overhead.

To support encryption of private objects, it becomes necessary to send a query twice the size of the original query statement given by the user. It is important to note here that the sub-stations and tags do not perform any of the encryption or decryption operations. They merely store information either in encrypted form or as clear text allowing them to be simple and cheap. Moreover, this provides a secure framework in which no exchange of keys over the wireless channel is required.

4. PROTOTYPE

To verify our hypotheses and explore the various issues in designing MAX, we implemented a prototype using Crossbow's MICA2 [7] motes. The mote platform features a 8-bit AVR microcontroller with 4 KB of RAM and 512 KB of EEPROM. TinyOS is used as the software platform. Wireless communication is achieved via Chipcon's CC1000 radio transceiver, using packets with maximum size of 29 bytes. The communication range is a few meters even at the lowest power setting.

The prototype is implemented over a single locality. A computer connected to a mote is used as both the query terminal and base station. A graphical user interface (GUI) is provided in the computer for the users. This prototype emulates the behavior of RFID tags and resource constrained sub-stations. The tags are assumed to be smart, i.e., having the ability to respond with their match counts for a query. The match count is the number of words in the query that exist in the descriptor of a tag.

The prototype protocol heuristically minimizes transmission cost and prevents overloading of sub-stations. Provision for reliable communication is also made. The limitations of the mote platform have also been taken into account.

4.1 Protocol

A user initiates a query by entering a query statement into the GUI at the query terminal. The sequence of events in the system is as follows.

1. On receiving the query from the query terminal, the base station makes an inventory of all the sub-stations around it before broadcasting the query statement to them.

2. Upon receipt of the query statement, each sub-station will discover all the tags around it by performing an inventory, as in the ISO15693 RFID [5] protocol. The query statement is then broadcast to the tags.

3. The tags compute the match count using the query statement. The sub-stations then poll the tags for their match counts. The sub-stations will subsequently poll for the tag descriptors.

Table 2: Time Taken to Locate Object

	Minimum	Maximum	Mean (Std. Dev.)
Manual	10 s	300 s	101.4 s (87.3 s)
MAX-aided	2 s	110 s	39.8 s (35.5 s)

4. Each sub-station returns a match count vector to the base station. The match count vector is a list of value pairs, with each pair consisting of a match count and the number of tags having that match count value.

5. From the match count vectors received from all sub-stations, the base station decides the match count threshold. Tags with match count above this threshold are selected and asked for their descriptors.

6. The base station polls the sub-stations for identities and the associated RSSI values of the selected tags. Tags which are previously retrieved via other sub-stations will be skipped, while the descriptors of the rest are downloaded sequentially.

7. The descriptors of the sub-stations are finally retrieved by polling. The result is compiled and delivered to the query terminal.

The selected tags are presented to the user in decreasing order of match counts, along with the sub-station of highest associated RSSI for each tag. The RSSI readings of the other sub-stations can also be retrieved by the user if required.

The protocol uses polling extensively to prevent collision and minimize delay. The use of polling also allows for missing packets to be detected and retrieved. Care is also taken to keep the volume of communication low by selecting tags based on the match count vectors and downloading the descriptors of selected tags only once.

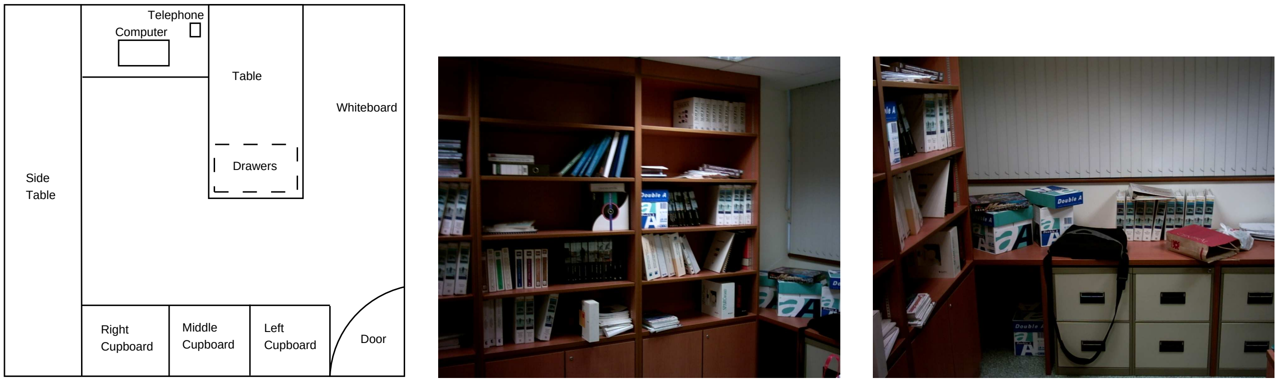
4.2 User Trials

To verify the hypothesis that landmark based localization is sufficient, we conducted a subjective user trial on the prototype system. The experiment was conducted in a 5 m by 5 m faculty office on our campus, as shown in Fig. 3. The room contained some wooden furniture and metal filing cabinets, and was cluttered with books and files to reflect a typical office environment. We tagged 4 objects in the office and placed 7 sub-stations.

Ten random subjects were invited to test the system and asked to find objects in the room. They were instructed to locate 2 objects with MAX and 2 objects manually. The subjects were given a simple description of what they were looking for, such as `Magazine:IEEE Communications, Aug 2002`. Similarly, the system also provided the users with only very simple and coarse description of the location, such as `side table`. At the same time, all the sub-stations and tags were hidden from sight, to prevent any extra visual cues.

There were 3 instances in which the users gave up on searching for an object without aid of the system, while with the system all users were able to find the objects they were looking for. Users said that the cues provided by MAX were useful in finding the objects. Numerical results on the time taken to locate the objects are summarized in Table 2.

We note here that the number of user trials were few and therefore meaningful conclusions cannot be drawn from the quantitative results. However, given the scope of the search



(a) Room Layout

(b) Middle and Right Cupboards

(c) Side Table beside Right Cupboard

Figure 3: Photographs of Office used for Verification Trials

(a mere 5 m by 5 m room), the qualitative results (e.g., 3 users gave up without the system) and the subjective feedback (users said they found landmark based information useful) received from the users, the trials validate our hypothesis that landmark based localization can be extremely useful to humans.

4.3 Prototype Design Experience

We will now share our design experience in prototyping MAX using the motes. We chose the motes since they are resource constrained wireless communication devices, similar to what might be used as sub-stations. Moreover, the motes can be readily integrated with Skye-Tek RFID readers [6] for future development. Due to the unavailability of RFID readers at the time of prototyping, we emulated the RFID tags using motes. Though this experience is likely to differ from an actual implementation of MAX, certain lessons can definitely be learnt.

Typically, RSSI values are assumed to be time varying and an unreliable aid in localization. However we found that the values returned by the motes during trials in a computer laboratory and a cluttered room very stable. In fact, we found that the tags were associated with the nearest 2 sub-stations with high probability during our user trials. One of the main themes of MAX is that humans are powerful sensors that can use approximate information to locate objects. From the fact that our prototype which only provides landmark based information based on RSSI has improved the subjects' ability to locate things, we can deduce that RSSI is good enough for landmark based localization.

Much unreliability in communication was experienced due to MAC collisions, resulting in lost packets. Such losses are not acceptable in our application and have to be handled with caution to ensure reliable information delivery to the user. We resolved this issue by using polling in our prototype. We note that this problem may be exaggerated since we emulated RFID tags using motes. In an actual implementation, RFID tags would not interfere with communications between base station and sub-stations.

We often experienced buffer overflows in the motes acting as sub-stations. To combat this, additional storage is required. If this is not feasible, the protocol has to be designed to minimize the buffer requirements. We opted to use the on-board EEPROM of the motes for additional storage. However this incurred large read-write times causing significant delays. The compromise between the cost of high speed memory and the delay experienced should therefore be borne in mind during future design.

5. OPTIMAL QUERY PROTOCOLS – METHODOLOGY

The experience with the simple prototype that was designed motivates us to investigate what the correct design choices are. The design choices available to us are in terms of devices (e.g., customized passive tags with comparators) and in terms of the querying protocols used. Two issues of critical concern are that of energy and latency. Since we assume that the sub-stations are battery powered, energy is a critical resource that should be conserved to ensure longevity of the devices. In addition, results must be returned to the user as quickly as possible to enhance user experience. To this end, we first describe a methodology which can be used to design delay optimal and energy optimal protocols. Using this methodology, we investigate the performance of the system with different device and protocol choices.

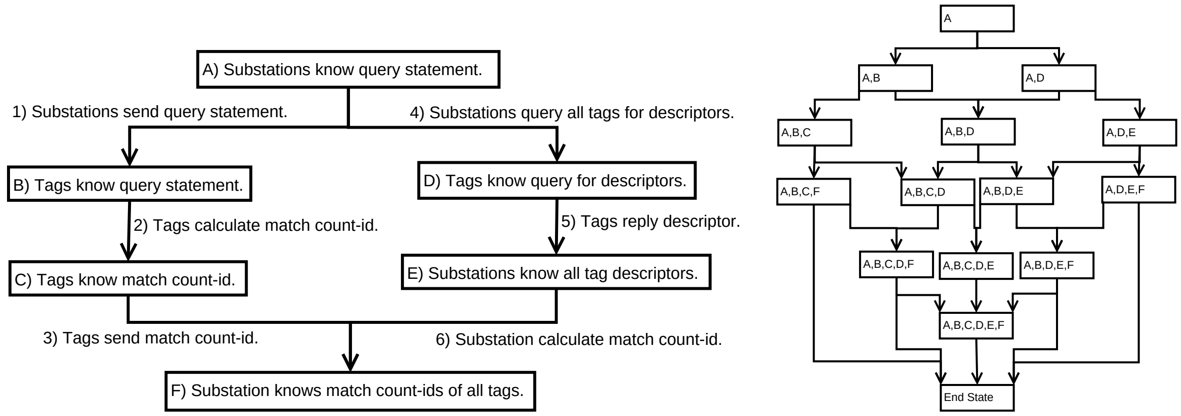
5.1 Problem Definition

A valid protocol is a set of actions, which brings the system from a starting state to a desired end state. We aim to find a valid protocol, which minimizes the energy consumed and/or latency incurred. Such a protocol is termed as the *optimal protocol*. The constraint of the optimization is the set of actions available to the system.

5.2 Methodology

To illustrate the methodology used, we will follow an example illustrated in Fig. 4. This is a small but representative subset of the final optimization performed, to find the optimal query protocols.

A heuristic manner of arriving at a protocol is often to think of the possible approaches. In Fig. 4(a), 2 possible approaches are shown. It is possible to make a choice



(a) Illustration of Possible Protocols

(b) States-Actions Graph for Optimization of Protocol

Figure 4: Optimization of Protocols

between these options, by comparing their total costs. However, hardly any optimality can be deduced for the resulting protocol.

To enable systematic search for the best protocol possible, we explore all possible combinations of actions. Each action will bring the state of the system from one to another. In addition, each action is associated with a cost, representing the effort required to execute it. Thus, the cost of a protocol is the summation of costs associated with all its actions. At the same time, each action requires a certain pre-requisite state before it can be performed.

To formulate a tractable problem, we try to ensure that the states are Markovian. This allows the possible actions at each state, to be independent of the previous states. We define the following terms.

DEFINITION 1. *A primitive state represents the knowledge a device possesses in the system. For example, primitive state A represents that the sub-stations know the query statement entered by the user, as shown in Fig. 4(a).*

DEFINITION 2. *An action transforms the system from one primitive state to the other and incurs a cost in the process. Given a primitive state-action pair, one can determine the subsequent primitive state. As illustration in Fig. 4(a), action 1 is described as "Substations send query statement". The action requires primitive state A as pre-requisite and bring the system to destination primitive state B.*

DEFINITION 3. *A state is defined as a vector of primitive states. For example, the state $[A, B]$ consists of primitive states A and B, as shown in Fig. 4(b). Thus, the system can be moved from state $[A]$ to state $[A, B]$, using action 1.*

If we appropriately define the states and actions for our system, we can then proceed to design an optimal protocol by finding the minimum cost path from the specified starting state to a desired end state. In our case, several end states can be accepted. Thus, we create an artificial end state which can be reached from the various possible end states using a costless action. In the example of Fig. 4(b), a costless action is defined for every state containing primitive state F to the arbitrary state *End State*.

Thus, we represent the states as vertices of a graph and actions as edges. Each edge has a weight corresponding to the energy or delay required to execute that action. We can now employ any shortest path algorithm, such as Djikstra's algorithm, to find the optimal protocol.

5.3 System Model

According to the above methodology, we formulated 45 primitive states and 64 actions, which are not listed here due to space constraints. The primitive states basically represent the knowledge of each device, which can be roughly categorized into query statement, match counts, descriptors and any decision made. Therefore, careful permutation of the possible primitive states can be performed. On the other hand, the possible list of processing actions is bounded by the information available at the device, i.e., the primitive states. One can only send the information to another device or process it to gain new information. Thus they can be exhaustively listed. It is essential to check that each primitive state must be reachable from another primitive state via an action unless it is the starting primitive state. At the same time, it has to have an action that allows it to move to another primitive state unless it is the ending primitive state. This will ensure that the set of primitive states and actions are consistent.

If one lists all the states possible, a total of 3.518×10^{13} states would be possible. By careful consideration, we can reduce the number of states drastically to $89,350$. This consideration is essential for the problem to remain tractable. Such consideration is demonstrated in Fig. 4, where a total of 32 states can be generated while only 14 are possible. The listing of states can be automated.

The hardware values used to determine the energy and latency costs of the various actions are listed in Table 3. These values are derived from currently available hardware, namely the Crossbow's MICA2 sensors, Skye-Tek's RFID reader and ISO15693 RFID tags.

We note that an inventory of the RFID tags by the sub-

Table 3: Hardware Device Parameters

Description	Symbol	Value
General Values		
Length of Match Value	L_M	1 byte
Length of RSSI Values	L_R	1 byte
Length of Query	L_Q	4 words
Maximum Length of Descriptor	L_D	8 words
Length of Each Word	L_W	16 bytes[a]
Base-station to Sub-stations Communication (MICA2DOT Mote)[b]		
Power Required to Transmit	P_S^T	8.7 mW[c]
Data Rate	R_S	38.4 kbps
Substations (MICA2 Mote)[d]		
Processing Speed	S_S	4 MHz
Power Required	P_S^A	46.5 mW
Substation-Tags Communication (SkyeRead M1-Mini[e])		
Power Required[f]	P_T	82.5 mW
Data Rate[g]	R_T	26 kbps
Tags (RFID)[h]		
Processing Speed[i]	S_T	20 kHz

[a]The average length of an English word, considering all words, is between 4 and 5. However, the average length of word in use may not be so. Thus, a convenient value is chosen for analysis before an more accurate value can be determined from further research.
[b]Information can be found in [8] and [9].
[c]This represents the additional power required for the sub-station to transmit. The power required to keep it awake is also required.
[d]Information are extracted from [8].
[e]The reader has a read range of 7 to 15 cm depending on the antenna used. The values are available in [10]. The SkyeRead M1-Mini is assumed to be switched off when not in use, thus drawing 60 μA.
[f]Power is provided by RFID reader.
[g]Data rate is based on ISO15693.
[h]Processing power would be the same as the communication power requirement at 82.5 mW, since the tags are passive.
[i]The processing of the RFID tag are accomplished by hardware implementations and thus support true parallel processing. However, the speed is highly restricted by the power available. Thus, a equivalent speed of 18.7 kHz is estimated from the required processing for comparing an unique identifier, as specified in ISO15693. Therefore, the convenient value of 20 kHz is used.

Table 4: Simulation Parameters

Description	Symbol	Value
Number of Substations		
Exact Number of Substations[a]	N_S	50
Number of Tags		
Sum of Tags under each Sub-station[b]	$\sum_i N_T^i$	2250
Maximum Number of Tags under a Substation	$\max_i N_T^i$	45
Number of Selected Tags		
Total Number of Selected Tags[c]	$N_{T(S)} = \sum_i N_{T(S)}^i$	40
Maximum Number of Selected Tags under a Substation[d]	$\max_i N_{T(S)}^i$	10
Scaling Factor for Different Decision[e]		
Decision by Substations		1.5
Decision by Base-Station, where a threshold match count is used for querying the descriptors.		1.3
Decision by Base-Station based on approximated statistics, where specific identities are used for querying the descriptors.		1.15
Decision by Base-Station based on accurate statistics and specific identities are used for querying the descriptors.		1

[a]We consider a 5 m by 5 m room densely deployed with substations. Thus, the total number of furniture and immobile objects is estimated to be around 50.
[b]An average of 30 tags is expected for each sub-station, giving a total of 1500 tags in reality. However, due to overlapping coverage of the sub-stations, the number is multiplied by 1.5. This degree of overlap is low, since we are considering Skye's RFID reader with a maximum range of 15 cm.
[c]We assume that a user would require a total of 40 results only. It is based on the intuition that searches are seldom useful beyond a small portion of the returned results.
[d]We assume an asymmetric distribution of desired objects in the room. This is assumed because it is common to cluster similar objects together.
[e]We note that the values of $\sum_i N_{T(S)}^i$ and $\max_i N_{T(S)}^i$ would be different for different types of decision made. For example in Figure , the tag d will be returned twice by sub-stations 1 and 2 if a local decision is made. This is because this overlap information is not available to all the sub-stations. Thus, we scale these values for different decisions made.

stations is a one-time process, which is of negligible cost to each query[2]. We also assume that the base-station is computationally powerful and line-powered. Thus, any computation within the base-station suffers negligible delay. We note that the minimization of energy is to achieve longevity, thus we neglect any energy cost of actions performed by base-station. We caution that communication between the base-station and sub-stations would incur cost at the sub-stations, which cannot be ignored. Therefore, the energy consumption of the protocols are only those incurred by the sub-stations and tags.

6. OPTIMAL QUERY PROTOCOLS – RESULTS

Armed with the above-mentioned technique and values in Tables 3 and 4, we compute optimal protocols for various scenarios, based on different choices available during system design. Firstly, we can explore the use of a customized smart tag that can calculate and transmit its match count to the sub-station. It can also transmit its descriptor according to a match count threshold. Next, we investigate the appropriate amount of processing to push down to different levels of the hierarchy. In a centralized system, the sub-stations are asked to report all responses or their statistics to the base-station. In a distributed system, each sub-station independently does some processing and decision making to decide what subset of the results or their statistics to be returned to the base-station.

We note that the results can be maximally relevant or heuristic[3]. A user would like to have $N_{T(S)}$ (as defined in Table 4) results for a single query. A maximally relevant result is one which consists of all of the $N_{T(S)}$ results with the highest match counts. This would require all match count and identity pairs to be returned to the base-station for selection to be performed. Alternatively, the base-station may decide on the match count threshold based on other statistics such as match count vector or a subset of the match count and identity pairs. In such cases, the results will be termed as heuristic.

Subsequently, we investigate the effect of overlap, which

[2]In the ISO RFID standard, the reader performs an inventory of tags in its vicinity. The tags are subsequently polled when they are read.

[3]Recall definitions from Section 2.2.

Table 5: Energy-Latency Values of Optimal Protocols

Scenarios[a]				Energy /J	Latency /s[b]
1	Centralized	Dumb Tags	Heuristic	12.346	4.672
2	Centralized	Smart Tags	Heuristic	0.663	3.355
3	Distributed	Dumb Tags	Heuristic	12.344	4.641
4	Distributed	Smart Tags	Heuristic	0.637	3.323
5	Centralized	Dumb Tags	Maximally relevant	12.511	7.762
6	Centralized	Smart Tags	Maximally relevant	0.767	6.378
High Overlap System					
7	Centralized	Dumb Tags	Heuristic	41.097	15.162
8	Centralized	Smart Tags	Heuristic	1.519	7.420
9	Distributed	Dumb Tags	Heuristic	40.810	12.674
10	Distributed	Smart Tags	Heuristic	0.860	3.764
Low Number of Tags					
11	Centralized	Dumb Tags	Heuristic	3.678	3.413
12	Centralized	Smart Tags	Heuristic	0.538	3.369
13	Distributed	Dumb Tags	Heuristic	3.672	3.283
14	Distributed	Smart Tags	Heuristic	0.538	3.283
High Number of Tags					
15	Centralized	Dumb Tags	Heuristic	17.782	5.729
16	Centralized	Smart Tags	Heuristic	0.757	3.658
17	Distributed	Dumb Tags	Heuristic	17.768	5.653
18	Distributed	Smart Tags	Heuristic	0.676	3.437

[a]Smart refers to the capability of the tags.
[b]If the optimal energy protocol is not equal to the optimal latency protocol, the energy of the optimal energy protocol and the latency of the optimal latency protocols are given.

is related to the number of tags a sub-station can see (or equivalently the number of sub-stations that a tag can communicate with). To test the effect of overlap, we computed the optimal protocols after changing the values in Table 4 to $\sum_i N_T^i = 7500$; $\max_i N_T^i = 225$ and scale $\sum_i N_{T(S)}^i$ and $\max_i N_{T(S)}^i$ by 5 for decision by match count decided by sub-stations and 4 for decision by match count decided by base-station[4]. These figures have changed to reflect the degree of overlap. Though the query optimization is independent of the devices used, it can be seen that different devices may have different overlap due to their communication ranges. Hence, degree of overlap will be one of the factors considered in device choices.

Finally, the number of tags is changed from 1500 to 500 and 5000, representing low and high number of tags respectively. The results are tabulated in Table 5.

6.1 Smart Tags

The optimal protocol for smart tags operates in multiple rounds. In the first round, tags which have matches to the query respond with their match counts. In the subsequent round, only tags which have match counts greater than some threshold are asked to respond with their descriptors.

From Fig. 5(a), we can observe a decrease in both energy consumption and latency when dumb tags are replaced by smart tags. The energy consumed decreased by an average of 17.66 times, while the delay incurred decreased by an average of 1.31 times for a low overlap system. We can therefore see an enormous benefit in deploying smart tags. We found that in all the optimal protocols for smart tags, the match count is tabulated by the tags themselves, which is not possible in dumb tags.

For a high overlap system, the energy consumed and latency incurred by using dumb tags is increased by 37.25 times and 2.71 times respectively, shown also in Fig. 5(a). Both values are significantly larger than that for low overlap systems. This signifies that the more overlap the system experiences, the larger the benefit of using smart tags.

Comment: We have shown that significant savings in latency and energy can be achieved with smart tags, especially in a high overlap system. Thus, the customization of tags should be seriously considered despite potential cost of losing the economies of scale.

6.2 Distribution of Computational Burden

To see how computational burden should be distributed, we compared the protocols for a centralized and distributed system. Comparing rows 1 & 3 and rows 2 & 4, it is somewhat surprising to observe little or no difference in energy consumption and latency between centralized and distributed decisions for a low overlap system. The energy consumed and latency changes by 0.990 and 0.996 times respectively.

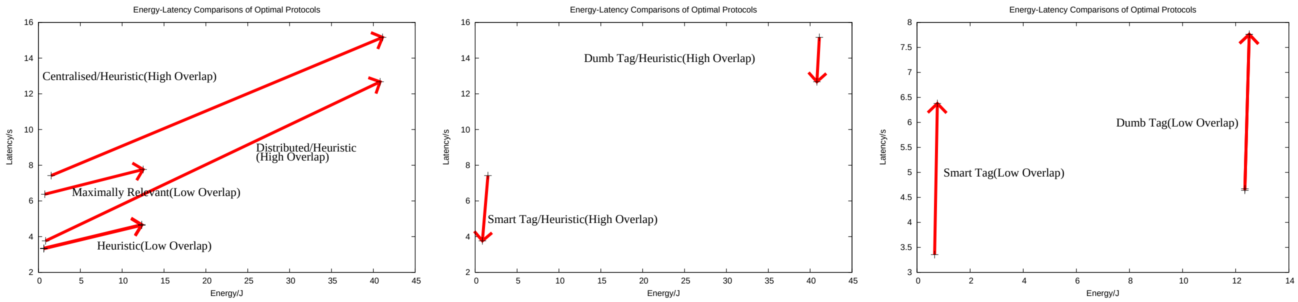
This result should be put into perspective. Firstly, the sub-stations are modelled as Crossbow MICA2 sensors, which consume large idle power. Thus, the energy for computation may well be significant compared to that for communication. Secondly, the degree of overlap of the system have significant impact on the results. As expected for high overlap system, significant benefits of distributing the processing can be seen in Fig. 5(b). The latency is decreased by 0.836 and 0.507 times for dumb and smart tags respectively. For smart tags, a significant decrease in energy consumed of 0.566 times can also be observed.

Comment: By pushing the computation load down the hierarchy, there is a slight improvement in a low overlap system, while significant benefits can be seen in a high overlap system. Therefore, the distribution of the computation load is more important as the degree of overlap increases in the system. This should be kept in mind during design.

6.3 Maximally Relevant Results

We would like to investigate the cost of ensuring maximally relevant results in a centralized system, as shown in Fig. 5(c). An increase in latency and little change in energy consumed is observed, when maximally relevant results are required over heuristic ones. They have increased by 1.789 and 1.097 times respectively. For maximally relevant results, the match count and identity pairs of the tags are all delivered to the base-station, accounting for the increased latency. However, as match count and identity pairs are small in size and querying by identities eliminates any

[4]We note that the exact number of selected tags will be transmitted if the querying is performed using identities decided by the base-station having a complete set of match count and identity pairs. However if querying is performed using a threshold match count decided at base-station, the descriptors of tags in the vicinity of more than one sub-stations will be retrieved multiple times due to overlap. Thus, the number of returned results must be scaled to reflect this condition. The condition is made worse by "ill-informed" local decisions performed at sub-stations.



(a) Smart Tags to Dumb Tags : Both energy consumption and latency increases when the tags are disallowed from calculating and transmitting their match count.

(b) Centralized Decision to Distributed Decision : When the substations in a high overlap system are disallowed from deciding which results are to be returned, the energy consumed decreases slightly while the delay is decreased significantly.

(c) Heuristic Result to Maximally Relevant Result : To ensure maximally relevant results, the energy consumed increases slightly but the latency increases drastically.

Figure 5: Energy and Latency Comparison of Optimal Protocols Set I : These plots are scatter plots of the optimal energy-latency values achievable under various scenarios (values provided in Table 5), where the arrows are drawn to show the trends under various changes.

unnecessary querying of descriptors, the energy increment is minimal.
Comment: We can see that a significant saving in latency can be achieved by foregoing maximally relevant results. However, the effect of providing heuristic results have to be investigated in an actual implementation. The aim is to provide sufficient information for the system to be useful, without violating the delay requirements.

6.4 Overlap

The effect of increasing overlap is investigated in Fig. 6(a). It can be seen that the increase in energy consumed ranges from 1.350 to 3.329 times. Similarly, the increase in latency ranges from 1.133 to 3.245 times. We can see a large increase in both energy consumed and latency incurred. The large increment in cost is due to the large number of repetitions, especially with dumb tags.
Comment: We can see an increase in latency and energy as the overlap of the system increases. Thus, the degree of overlap in the system must be carefully controlled, while providing coverage within the locality.

6.5 Number of Tags

We look at the effects of increasing the number of tags. The results are shown in Fig. 6(b). As the number of tags increases from 500 to 1500, the energy consumed and latency incurred increases by 2.284 and 1.198 times respectively. An increment of 3.084 and 1.383 times is observed respectively in the energy consumed and latency incurred as the number of tags is increased from 500 to 2500. We can see that the increase is marginal for smart tags, allowing for more tags per sub-station to be deployed.
Comment: The latency and energy increases as the number of tags increases. This would place a limit on the number of tags that can be served under a base-station, given a certain delay constraint. This points to the need for multiple base-stations to serve large locality.

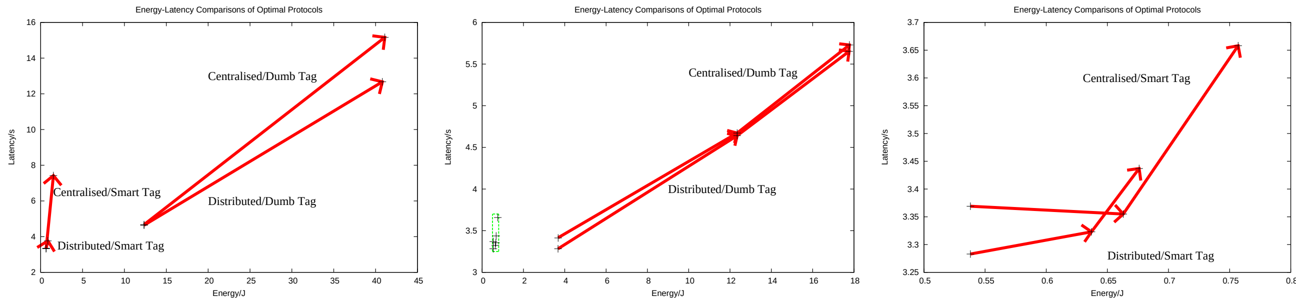
7. RELATED WORK

Localization algorithms that have been developed for indoor and urban environments are described and summarized in Table 1 of [11]. The modalities involved are usually radio frequency, ultrasound and video capture. These localization algorithms can be classified as being either location tracking systems or location support systems. More notable recent developments include the Ubisense[5] [2] location tracking system and the Cricket [4] location-support system. These systems use ultrasound, RF and ultra-wideband techniques to do accurate localization via carefully placed beacons. A comparison of the RF-based systems can be found in [12].

The Ubisense system proactively keeps track of the locations of objects and stores them in a central database. A context aware middleware enables a variety of smart space applications. It requires careful placement and calibration of what are called sensors (about 4 for every 400 square meters) so that all tags can be localized accurately within a physical space. Since all object locations are tracked proactively and stored in a central database, it can provide landmark based localization quite easily. However, since object locations are stored centrally, it does not provide for either privacy of objects or physical spaces. Also, based on our experience, it is difficult for a lay person to set up the system easily.

Cricket offers a location-support system, as opposed to a location tracking system. It uses a set of carefully placed beacons to estimate positions accurately. In Cricket, the devices estimate their own locations and broadcast them as needed. Locations are not stored at a central location ensuring one aspect of privacy. However, in Cricket there is no provision to distinguish between private and public objects or private, off-limits and public spaces. In addition, since it requires careful placement of beacons, it is not easy to configure. Finally, in order to give human-centric location

[5] Ubisense is the commercial product developed from the Active Badge and Bat projects.



(a) Increasing Degree of Overlap Both energy consumption and latency increases with increased overlap, especially significant in systems using dumb tags.

(b) Increasing Number of Tags : As the number of tags increases, the energy consumption and latency of systems using dumb tags increases. See Figure 6(c) (expanded version of the bottom left of this plot marked by a green box) for results on systems using smart tags.

(c) Increasing Number of Tags (Smart Tags) as expanded from Figure 6(b) : As the number of tags increases, the energy consumption and latency of systems using smart tags increases less significantly as compared to those using dumb tags.

Figure 6: Energy and Latency Comparison of Optimal Protocols Set II : These plots are scatter plots of the optimal energy-latency values achievable under various scenarios (values provided in Table 5), where the arrows are drawn to show the trends under various changes.

information it requires an overlay of map servers, implying that it is not robust to reconfigurations of the physical spaces.

There are also other RFID-based systems designed for urban deployment. The Bewator CoTag [1] technology is mainly designed for security monitoring and control, while Wavetrend [13] RFID systems are targeted at pharmaceutical, heath care, manufacturing and warehouse management. Such system requires experts for deployment. Moreover, they do not provide privacy and are also not robust to reconfigurations in the physical space.

Localization has also been studied in the context of wireless sensor networks. There is an arsenal of algorithms proposed for these purposes. In [14], the schemes are divided into range-based and range-free. The common modalities used, in both categories, are radio frequency or ultrasound. These works aim to provide exact localization of the nodes or to place them in relative positions to each another. Often, beacons whose locations are known via Global Positioning System (GPS) or other means, are required [15–17]. Clearly such algorithms are not suitable for the application we have in mind.

There are also projects like the NIST Smart Space [18]. This project aims to create a smart environment which predicts and reacts to the needs of individual users. However, these projects hold no notion of localizing objects. They concentrate mainly on the communication of information.

Some may relate this work to sensor database systems, such as Cougar [19] and TinyDB [20], since we query the physical environment. In fact, our work does share similarity in design of a query process, that utilizes "in-network" processing. However unlike these systems, we are not dealing with sensor data. The output of our tags are generally non-redundant and thus cannot be aggregated. Moreover, localization is not of concern in these systems.

8. REFLECTIONS

In this paper, we proposed a system that allows human-centric search of the physical world. Specifically, the system was designed with the goal of (i) human-centric operation, (ii) privacy and security and (iii) efficiency and scalability. We proposed a hierarchical (possibly hybrid) architecture which exploits the fact that humans are powerful sensors who require only visible cues and landmarks to locate and search for objects. Our architecture allows for two kinds of privacy, namely spatial privacy and object privacy. It is interesting to note that the notion of privacy and security in this system is different and arguably more complex than that in traditional systems in the digital world. Finally, we investigated the design of optimal query protocols and showed that with a slight customization of passive RFIDs, significant improvements in energy and latency could be achieved.

Most importantly, the design of MAX allows for the paradigm of "everything has *a* place" rather than "everything has *its* place". This, combined with the new notions of privacy and security provided by MAX, will enable a variety of new applications which involve the sharing and trading of physical resources.

From our results and experiences, we have gained valuable insights into some overarching issues in the design of the system. A review of these is provided in the following.

- *Use of RSSI:* In this paper we have used RSSI to estimate approximate location of an object. In localization literature it is usually argued that RSSI is not a good metric due to the time varying nature of the wireless channel and the effects of shadowing. However, in this paper we argue that for the purpose of our system, RSSI is sufficient.

  First we consider largely static environments (such as offices and homes), therefore we can assume that channel fluctuations are small. Second, due to shadowing effects, an object might not be associated with the nearest sub-station. However, in our system we argue that a transmission range of about one meter should be used for the sub-station. This will imply that the localization error is always bounded within a circle of radius one meter. Since we are attempting landmark based localization, this will be sufficient to help a human locate an object quickly.

Finally, there is the issue of locating objects which are encased in metal (e.g., a metal cabinet). In such cases, when the sub-stations are placed/embedded in the cabinet, it can be designed in a manner such that the antenna is on the outside of the metal cabinet (this could be done at the time of manufacture, taking aesthetic considerations into account). This will allow objects enclosed within metal cabinets to be located.

- *Choice of Tags:* It is clear that the choice of devices for the tags is of paramount importance. If devices like smart dust or active RFID are employed, they would be capable of more processing since they are battery powered. The fact that we see significant improvements in energy and latency by adding a limited processing capability to passive tags seems to indicate that additional intelligence at the tags is desirable. However we still need to investigate if there is a cost versus performance benefit trade-off to using active RFID or smart dust like devices.

- *Smart Tags:* We have investigated customized passive tags (defined as smart tags) which allow them to respond conditionally to queries. We point to the fact that these operations can be achieved through simple logic gates, summers and comparators. This ensures that the power delivered by the RFID reader is sufficient for execution of these operations. Currently, ISO15693 standard RFID tags already have the ability to compare their identities and conditionally respond in the inventory phase [5]. The main issue in having smart tags is the customization. Customization may lead to a loss in economies of scale resulting in a trade-off between the performance and cost of the system.

- *Energy/Latency Trade-off:* Another interesting aspect is the trade-off between energy efficiency and latency. In most wireless systems, improving the performance of one metric implies relaxing the requirement on the other. However, due to the nature of passive RFID, this is not true for most cases. In our design of optimal protocols, we found that in most cases the delay optimal and energy optimal protocols are identical.

In this paper we started with objective of designing a human-centric search system for the physical world. We observed that humans are powerful sensors with comprehensive cognitive and computational capabilities. We posited that this observation could be exploited to design a simple, efficient and scalable search mechanism. We conducted user trials of a simple prototype system in a small 5m by 5m mildly cluttered office space. The subjective user feedback was positive and extremely encouraging, validating our simple yet powerful hypothesis.

9. ACKNOWLEDGMENTS

The authors would like to acknowledge Chern-Sia Lim, Phillip for his work in developing the prototype and all the anonymous reviewers for their helpful comments. Finally, we would like to express many thanks to our shepherd Deborah Estrin for her valuable comments which helped to improve the paper.

10. REFERENCES

[1] (2005, April) Bewator uk. [Online]. Available: http://www.bewator.com/uk/index.xml

[2] (2005, April) Ubisense-the smart space company. [Online]. Available: http://www.ubisense.net/

[3] Kumar, S., Alaettinglu, C. and Estrin, D., "Scalable object-tracking through unattended techniques (SCOUT)," in *Proc. International Conference on Network Protocols*, 2000, pp. 253–262

[4] A. N.B.Priyantha and H.Balakrishnan, "The cricket location-support system," in *Proc. 6th Ann. Int'l Conf. Mobile Computing and Networking, (Mobicom 00)*, 2000, pp. 32–43.

[5] *ISO/IEC 15693 Identification cards - Contactless integrated circuit(s) cards - Vicinity cards*, 1st ed., ISO.

[6] (2005, April) Skye-tek. [Online]. Available: http://www.skyetek.com/readers_Mini.html

[7] (2005, April) Crossbow technology - inertial & gyro systems, wireless sensor networks, smart dust, advanced sensors. [Online]. Available: http://www.xbow.com/

[8] "Power technology and management," www.xbow.com, presentation from Wireless Sensor Networks: A Two-Day Hands-on Seminar, June 2004.

[9] "Radio, rf concepts, and tos radio stack," www.xbow.com, presentation from Wireless Sensor Networks: A Two-Day Hands-on Seminar, June 2004.

[10] *Product Datasheet, SkyeRead M1 Mini*, SkyeTek, www.skyetek.com, December 2004.

[11] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, pp. 57–66, August 2001.

[12] P. Steggles and J. Cadman, "A comparison of rf tag location products for real world applications," Ubisense, Tech. Rep., March 2004.

[13] (2005, April) Wavetrend. [Online]. Available: http://www.wavetrend.net/

[14] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Mobicom'03*, September 2003.

[15] F. Mondinelli and Z. K. Vajna, "Self localizing sensor network architecture," in *Proceedings of the 19th IEEE Instrumentation and Measurement Technology Conference*, vol. 1, May 2002, pp. 823–828.

[16] P. Bergamo and G. Mazzini, "Localization in sensor networks with fading and mobility," in *IEEE PIMRC*, September 2002, pp. 750–754.

[17] A. Tarnacha and T. L. Porta, "E-STROBE: An adaptive beacon activation algorithm for sensor localization," in *Proceedings of VTC*, 2003.

[18] (2005, April) Nist smart space laboratory. [Online]. Available: http://www.nist.gov/smartspace/

[19] (2005, April) COUGAR: The network is the database. [Online]. Available: http://www.cs.cornell.edu/database/cougar/

[20] (2005, April) TinyDB a declarative database for sensor networks. [Online]. Available: http://telegraph.cs.berkeley.edu/tinydb/