Development of a Wireless Location System in Lindley Hall*

Rahul Doshi Tom Jagatic Ashraf Khalil
Satoshi Shirasuna Poornima Venkatakrishnan

January 6, 2004

Abstract

Development of context-aware applications is a growing area of research in pervasive computing. Determining nearby resources or object tracking are few examples of utility. It is for this reason a wireless networking location system for Lindley Hall was developed. In comparison with other location systems such as Cricket and Bat, it was deemed a model similar to the RADAR system was best suited given available resources and cost. The original RADAR specification only took into consideration signal strength measurements in 2D-space, whereas our prototype extends the implementation to 3D-space. Signal strength measurements from the first floor of Lindley Hall were taken to build a radio map used for location prediction of a mobile station. The results of testing accuracy were comparable to the RADAR implementation and encourage further work in this implementation.

1 Introduction

1.1 A Wireless Network Location System

The consideration to develop a wireless network location system using signal strength (SS) and/or signal to noise ratio (SNR) measurements from existing radio frequency (RF) networks to predict the location of objects is not a novel concept. Numerous implementations have been developed using this model. Our model extends one of the first–the RADAR[8, 9] model by taking into consideration AP measurements from 3D-space (above and below floors of a building) as opposed to 2D-space implementation.

Wireless access points (APs) typically provide overlapping coverage among an area of service. The fundamental principle is the strength of the RF signal is a function of the distance from the AP to the mobile station (MS). The closer

*Facility of the Indiana University Computer Science Department.

the MS is to the AP, the stronger the signal is. Likewise, the fewer obstructions by line-of-sight between the AP and the MS, the stronger the signal will be.

The RADAR model uses the following approach to determine MS location. A radio map is created by taking measurements of SS from multiple overlapping APs. The data is stored in a database in the form of (x, y, z, ss_i), where (x, y, z) are the coordinates of the location where the SS is recorded and ss_i is the SS at the i^{th} AP. Multiple measurements are taken at the same location and at differing times and with different orientations in order to have a more accurate sampling.

1.2 Goal

The focus of this project is to investigate the feasibility of deploying a wireless location system in Lindley Hall[13]. The APs of Lindley Hall are deployed in such a manner by design that they cover 3D-space rather than 2D-space investigated in the RADAR project. This 3D-space implementation provides an interesting area of research in development of a wireless location system.

1.3 Project Milestones

A number of significant milestones identified in the project proposal included the following tasks.

Data Collection Code which invokes the methods that provide the SS readings from the device (wireless card) driver/API must be written. This code also stores these readings in a way that provides easy and fast access to them at later times. This can be a special purpose data structure, database, or simply a file. Deciding what kind of wireless card and operating system (Windows or Linux) plays an important role since the OS may dictate the ease of collection.

Constructing the Radio Map After creating the data collection code, a radio map for the building must be built. Constructing the radio map requires deciding the granularity of the readings (the distance between every reading), how many readings are needed to obtain a realistic average, and what is the best time to do the measurement (nighttime, daytime, or both).

Design of a Prediction Algorithm Taking into consideration the observed SS measurements by location and orientation, a prediction algorithm must be constructed which accommodates SS variation as well as ommission of AP measurements.

Testing the System Once the radio map is created and a preliminary prediction algorithm designed, experiments can be performed to test whether the location system provides acceptable measurements. Questions such as, what is needed increase the precision (more APs, more readings, different radio maps for different times, environmental profiling) can be considered.

Development of Tracking Application Lastly, a sample application which tracks a MS and identifies position on a map that represents Lindley Hall. The application can be deployed on a hand-held device that has a wireless connection. This application can be extended to show more information such as nearby printers or other services in the department as the user gets closer to them.

2 Tools and Infrastructure

The system developed makes use of the following environment.

Hardware

1. IBM Thinkpad Laptop, Windows XP
2. Orinoco Gold wireless network card
3. Agere/Lucent wireless APs
4. Sun Blade server, Solaris 2.8

Software

1. Visual C++/Visual Studio .NET
2. Apache, Perl, ImageMagick, MySQL, CVS
3. Wireless Research API (WRAPI)
4. Wireless Mobile Received Signal Strength Indication (WMRSSI) Library

Measurement Tools

1. Three tape measures (1–30 foot, 2–12 foot)
2. One dry erase cleaning bottle (used as ad-hoc plumb bob)
3. Dental floss, Post-It notes, and Sharpie markers

WRAPI Overview

WRAPI 2.0[5] consists of a set of wireless LAN monitoring tools built for Windows XP-based systems. The WRAPI software library allows applications running in user-space on mobile end stations to query information about the IEEE 802.11 network they are attached to. WRAPI works with any IEEE 802.11b wireless network hardware vendor.

WRAPI functions obtain information about the wireless LAN using the Network Driver Interface Specification User Mode I/O (NDISUIO) Protocol.

NDISUIO is a connection-less, Network Driver Interface Specification (NDIS) 5.1 compliant protocol driver. It allows user-mode applications to establish and tear-down bindings to network adapters (Ethernet, WLAN etc.) Further, it also supports setting packet filters, sending and receiving data, and handling plug-and-play events. Therefore, as an NDIS aware component, NDISUIO can directly open an NDIS miniport driver (i.e. network card driver) to send requests, set, and query information. NDISUIO provides an interface between a user-mode application and NDIS using DeviceIoControl (similar to the Unix ioctl).

3 Measurement

Measurements were done with a team of four. Work was split evenly for laying out the tape measures, marking the X, Y and Z coordinates on the Post-It notes and taking SS measurements using the IBM Thinkpad.

To perform SS measurements each room was given a unique identifier[1] and sectioned into $5x5$ foot cells. The entire first floor was divided in multiple coordinate frames; each room acting as one coordinate frame. The northeast corner of every room was selected as the origin with east direction as the x-axis and north direction as the y-axis. Ten SS samples were taken for each location and orientation at 500 ms intervals to take into account variation in SS. All the measurements were done with the Thinkpad kept at the waist height (Z-coordinate of 44 inches) except for few measurements in LH120. For uneven surfaces like in LH102 height was adjusted to keep the relative height from ground level constant to 44 inches.

Eighty-six percent of the time, measurements were taken between the hours of 7 P.M. and 9 P.M. It is observed from Figure 2 signal strength varies greater during this time.

Initial measurements to build the radio map were done using WRAPI but the system was tested using WMRSSI library. Comparisons were performed to confirm WRAPI and WMRSSI provided the same SS information by measuring the SS from both for a particular location and orientation. Received SS measurements were found to be almost identical in both cases.

To construct the radio map for the first floor, the measurement process took approximately 15 hours over the period of two weeks.

Problems Encountered

Considerable time was spent porting WRAPI to work with the building wireless LAN due to limited documentation and other software dependencies. It seems intuitive having more parameters, such as SNR, would have increased the accuracy of location prediction. There is no device independent way to get SNR,

[1] Identifiers were chosen mostly by names of the rooms except for the corridor which was named as LH1FC (Lindley Hall First Floor Corridor).

however. Therefore, our implementation using Windows did not focus on using SNR.

The IWSPY[6] tool in Linux provides SNR in addition to SS information. Measuring SS information in Linux using IWSPY wasn't practical for our system due to the fact that there was significant delay before the SS readings from the APs was updated.

In certain areas of the Lindley Hall it is difficult to sample SS information. No measurements were taken in portions of LH135 due to physical obstructions like cables and other un-movable hardware. Also, measurements were not taken in the restrooms, hub closet, nor custodial closet.

4 Data Manipulation

Creating an effective radio map is the most important part in deploying a wireless location system using this technique. As expressed in section 3 it was deemed necessary to perform multiple SS samplings and calculate an average of these samplings to account for variation.

Table 1: Raw measurements collected. A description of the columns from left to right: iteration number; date; time in GMT -0500; user performing the collection; location identifier; x-coordinate; y-coordinate; z-coordinate; direction of user (N,E,S,W); MAC address of AP; and SS in dBm.

#	Date	Time	User	LocId	X	Y	Z	Dir	AP	SS
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:52:6a:5a	-76
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:50:96:26	-83
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:01:ac:3f	-84
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	**00:02:2d:b1:14:dc**	**-91**
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	**00:02:2d:b1:14:dc**	**-92**
1	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:60:1d:f0:ed:34	-94
2	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:52:6a:5a	-76
2	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:01:ac:3f	-81
2	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:02:2d:50:96:26	-86
2	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:60:1d:f0:ed:3c	-89
2	10/28/2003	19:36:49	tjagatic	LH115	20	144	44	S	00:60:1d:f0:ed:45	-91
3	10/28/2003	19:36:50	tjagatic	LH115	20	144	44	S	00:02:2d:52:6a:5a	-73
3	10/28/2003	19:36:50	tjagatic	LH115	20	144	44	S	00:02:2d:01:ac:3f	-78
3	10/28/2003	19:36:50	tjagatic	LH115	20	144	44	S	00:02:2d:50:96:26	-84
3	10/28/2003	19:36:50	tjagatic	LH115	20	144	44	S	00:60:1d:f0:ed:45	-89
3	10/28/2003	19:36:50	tjagatic	LH115	20	144	44	S	00:60:1d:f0:ed:3c	-89
...										

A big step in construction of the radio map was determining what data to include and what to exclude. Preliminary assessments immediately following data

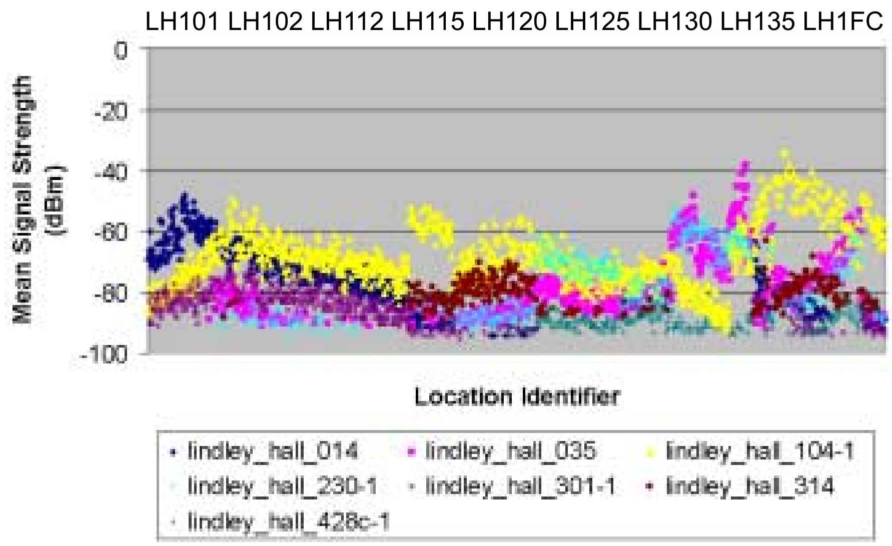

Figure 1: Processed signal strength by location identifier. It can be observed graphically unique patterns of AP signal strength exist.

collection for a given location indicated a number of interesting observations–some intuitive and some not. Table 2 illustrates the observations made.

There are seven known APs in Lindley Hall. Our data collection captured twelve APs. It is speculated these unknown APs were potentially in adjacent buildings and/or unaccounted/rogue APs in Lindley.

Another interesting observation entailed duplicate SS measurements for the same iteration/same sampling[2]. As mentioned earlier, for each location and orientation, ten iterations of SS samplings were performed at 500 millisecond intervals. Should duplicate readings persist in the mean SS calculation, it would be possible to calculate the mean of greater than ten iterations, yielding a bias to the duplicate readings. This concern was addressed, by discarding the duplicate readings.

When collecting data it was not atypical for access points to fade in/out, especially those with weaker signal strengths. Of concern was determining a mean SS with fewer than two iterations observed. In such occurrences these samplings were discarded.

Lastly of concern were outliers. An acceptable range in standard deviation was experimentally determined to be 5.0 dBm. If the standard deviation of a series of ten (or fewer) iterations exceeded this range, an outlier was determined and then discarded. This process repeated until the standard deviation was less-than or equal-to 5.0 dBm. Determination of an outlier was performed by

[2]Note in Table 1 duplicate SS measurements for access point, 00:02:2d:b1:14:dc.

calculating the median of a series of iterations, and then choosing the sampling furthest away from the median.

Table 2: Signal strength measurement observations.

	Uncleansed Data
40,510	SS samplings
12	Observed APs
	Discarded Data
3377	Duplicate SS samplings or < 2 iterations
151	Outlier SS samplings
	Cleansed and Processed Data
36,982	SS samplings
7	Known APs
259	Distinct x,y,z positions
3999	Position–orientation–AP SS measurements
1005	North position–AP SS measurements
1012	East position–AP SS measurements
994	South position–AP SS measurements
988	West position–AP SS measurements

The cleansing and processing[3] of data was performed using the Perl programming language. Each iteration sampling was put into a hash with a key comprised of the location identifier; the x,y,z coordinate; the orientation; and the access point. The value of the hash for a given key was the series of SS measurements. The hash was built in this manner by reading the raw data from standard input[4]. Checks were used when building the hash to eliminate duplicate readings in the same iteration and also ensure a hash once populated was not overwritten.

The second step of processing involved stepping through the hash and calculating the mean SS by location identifier; x,y,z coordinate; orientation; and access point. If a hash key had fewer than two samplings, the results were discarded. Likewise in this step, the standard deviation was calculated and outliers determined and discarded from the mean calculation.

The result of the above processing yielded the radio map used for location prediction.

[3]Cleansed data refers to data in which unknown and duplicate samplings have been removed; processed data is the mean of ten or fewer iterations (with outliers removed) calculated for a given position–orientation–AP.

[4]The collected data were stored in a comma delimited file identical in structure to Table 1.

5 Data Analysis

5.1 Prediction Algorithm

The prediction algorithm is used to determine the location of the MS, using the radio map. The processed data of the previously measured signal strengths were stored in a database which serves as the radio map. The radio map is organized as follows:

LocId	X	Y	Z	Direction	AP	SS
LH101	160	225	44	W	00:60:1d:f0:ed:34	-82

Here LocId denotes the Room Number on the first floor of Lindley Hall where this corresponding reading was taken. The X and Y coordinates denote the position inside the room (in inches) and Z denotes the height(in inches) at which the laptop was held when the readings were taken. Direction denotes the orientation i.e., East, West, North or South. The column AP denotes the MAC address of those APs which responded to the network card at the location and finally SS is the strength (in dBm) of the signal from that particular AP.

The input to the prediction algorithm is a set of pairs of MAC addresses and SSs returned by the wireless NIC at the test location on the first floor of Lindley (Table 3). This indicates the wireless network performance at the test location.

Table 3: Input example

AP	SS(dBm)
00:02:2d:52:6a:5a	-46
00:60:1d:f0:ed:45	-55
00:02:2d:01:ac:3f	-81
00:60:1d:f0:ed:34	-85

The algorithm receives this input and predicts a location for this input as follows:

1. Check if the input AP is in Lindley Hall and filter out the alien APs and their corresponding SSs.

2. Check for weak signals in the input, where the threshold SS value to determine a weak signal is decided by the parameter W. All signals with values less than W are considered weak. Hence now the input to the algorithm is a subset of the original input with weak signals removed.

3. The algorithm collects all records from the radio map which have the same MAC addresses as in the input and with their SSs such that $SS - X < SS < SS + X$ where X is the parameter to decide the SS range window

size. The records from the radio map which satisfy the above conditions are inserted into a temporary table for further processing to determine the most accurate prediction.

4. The resulting records from the previous processing contain not only the AP and SS information, but also the LocId, X,Y,Z coordinates and direction since they were retrieved from the radio-map with the structure described earlier. These records are further filtered and the number of candidate points narrowed down as follows:

   All those records which have

   (a) Common location(i.e., same LocId,X,Y,Z,Direction) at which all the APs in the Input set are active

   (b) Their corresponding SSs within $SS - X$ and $SS + X$ range. This again reduces the number of possible locations and retrieves a subset of the records obtained in the previous step.

5. To select the best result from the possible locations returned by the previous step, distance calculation measures such as Euclidean and Manhattan distance were used.

   Euclidean distance is calculated as the following:

   $Error = Sqrt((SS_1 - SS'_1)^2 + (SS_2 - SS'_2)^2 + (SS_3 - SS'_3)^2 + ...)$

   and Manhattan distance is calculated as the following:

   $Error = |(SS_1 - SS'_1)| + |(SS_2 - SS'_2)| + |(SS_3 - SS'_3)| + ...$

   Where SS_1, SS_2, SS_3 ... are the SS values in the input set and SS'_1, SS'_2, SS'_3 .... are the SS values of predicted location.

   Error value denotes the distance between the actual location and predicted location. Hence the location with minimum error is determined as the predicted location. In comparison of performance with Euclidean and Manhattan distance we found that both performed similarly. Hence we decided to use Euclidean distance for further performance analysis.

5.2 Testing

In order to determine the extent different factors (i.e. orientation, effect of humans, time, time intervals and number of iterations effect SS), analysis of SSs were done in different configurations. The observations are listed below:

1. When tested with the same person measuring SSs for the same location but in different directions, it was observed that there was a considerable difference in the SSs. This trial was done at LH125 at 60,24,44 coordinates. In order to interpret this statistically, the SS values on all 4 directions were averaged and the standard deviation of the individual values were determined. Table 4 shows the results. It indicates that there can be considerable variation in SS due to change of orientation.

Table 4: Testing for different location

PERSON	LocId	X	Y	Z	Direction	SD
Rahul	LH125	60	24	44	N	1.745
Rahul	LH125	60	24	44	S	1.777
Rahul	LH125	60	24	44	E	2.370
Rahul	LH125	60	24	44	W	1.861

2. When tested with different people measuring the SS at the same location and same direction, it was noticed that SS seemed to vary considerably even then. Measurements from 4 individuals were averaged and standard deviation (SD) readings from each individual were determined. This observation is included in Table 5.

Table 5: Testing with different people

PERSON	LocId	X	Y	Z	Direction	SD
Rahul	LH130	22	48	44	N	2.530
Tom	LH130	22	48	44	N	2.068
Satoshi	LH130	22	48	44	N	2.577
Poornima	LH130	22	48	44	N	2.586

3. Another test performed was the comparison of a single input reading and multiple iterations averaged. In single reading processing, at a fixed time we measured the MAC and SS values just once for a location and attempted to evaluate the location predicted by the measurement. For the latter method, we measured the SSs for a given location with 10 iterations with a time interval of 200 milliseconds. Then this was input to the algorithm and the location predicted. We compared the results obtained from both methods and it was observed when the input was an average of multiple values, the performance was better than just measuring the SS once.

4. When testing with different time intervals between iterations measuring SS (using the multiple iterations method described earlier), it was observed with an interval of less than 200 milliseconds there was no variation in SS for 10 iterations. But when the time interval was increased to 200 milliseconds or greater, an apparent change in the SS by +/- 2 dBm was observed.

5. Finally testing was done to observe the change in SS for the same location over 24-hours, so that during this period the effect of atmospheric conditions, human presence, etc. on the SS can be observed. A laptop with the

SS measurement program measuring the SS for 24 hrs at time intervals of one minute was set up at LH130 at 300,100,44 coordinates facing north. Figure 2 illustrates the variation in SS.

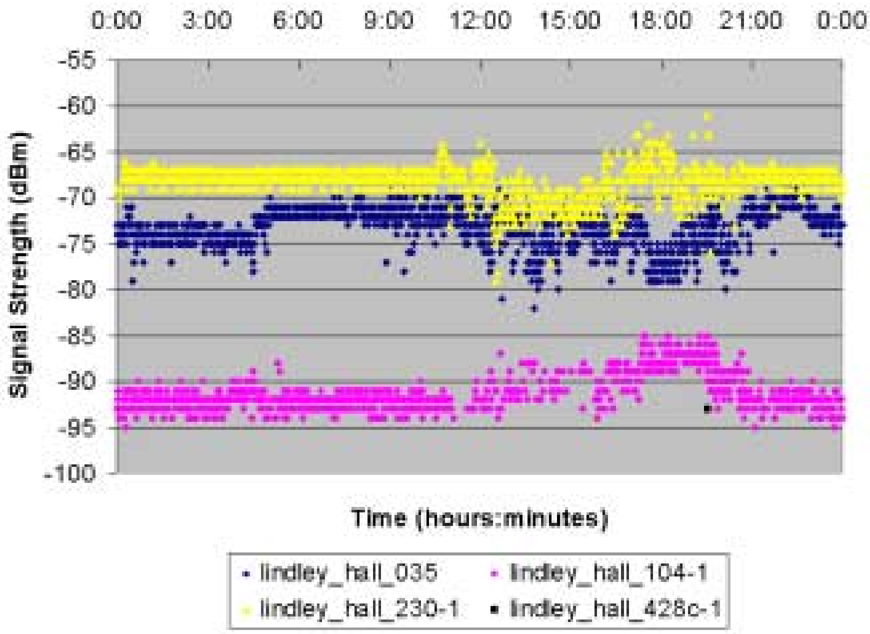

Figure 2: 24 hour analysis of signal strengths

In Figure 2 it can be seen that the stronger signals from APs lindley_hall_230-1 and lindley_hall_035 communicate with the MS throughout the 24-hour analysis. Although, the AP lindley_hall_104-1 has a weaker signal, it is also observed to steadily communicate with the MS. There is also another AP lindley_hall_428c-1 which has been active only once during the entire analysis process and it was observed to respond with a very weak SS. The graph shows the variance of SSs for the 3 APs which responded frequently over the 24-hour period. A statistical analysis of the SSs collected is included in Table 6.

5.3 Performance Analysis

The performance of the prediction algorithm was studied by varying the W and X parameters (discussed in the previous section). The total number of test locations used for analysis was 70.

Table 6: 24 hour analysis of signal strengths

AP advertised name	Average SS(dBm)	Maximum SD of SS (dBm)
lindley_hall_230-1	-68	1.732
lindley_hall_035	-73	2.111
lindley_hall_104-1	-91	2.011

5.3.1 Accuracy: Analysis of Prediction of Correct Rooms

Figure 3 illustrates the number of correct predictions with room-level granularity. This was done for parameter X with values 5 inches and 6 inches and for parameter W with values -80, -83, and -86 dBm. The observations from Figure 3 are summarized in Table 7.

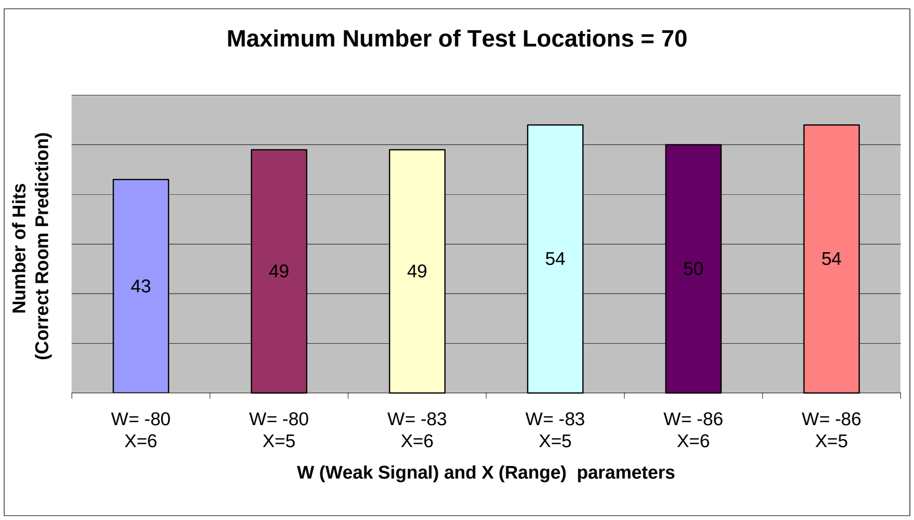

Figure 3: Analysis of prediction of correct rooms

Hence this indicates that the performance was best at 2 conditions

1. when X had a value of 5 inches and W with -83 dBm
2. when X had a value of 5 inches and W with -86 dBm

Both these conditions predicted 54 of 70 test locations to be in the right room. This indicates that we achieve an accuracy of 77% with room-level granularity.

Table 7: Analysis of prediction of correct room

Weak signal Threshold (W) (dBm)	SS Range Window size (X) (dBm)	Number of Correct room Prediction out of 70 Locations
-80	12 ($\pm$6)	43
-80	10 ($\pm$5)	49
-83	12 ($\pm$6)	49
-83	10 ($\pm$5)	54
-86	12 ($\pm$6)	50
-86	10 ($\pm$5)	54

5.3.2 Precision: Analysis of Average Distance of Predicted Location to Actual Location

To perform this, the individual origins for each room were removed and all points were mapped to a single origin in the north-east corner of the first floor of Lindley Hall. Figure 4 shows the average distance between the location predicted by the algorithm and the actual location for the 70 test locations that were considered.

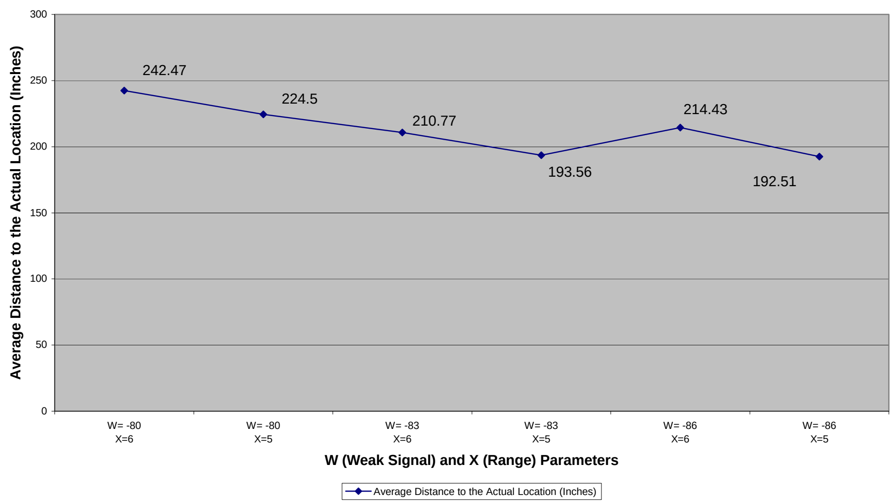

Figure 4: Analysis of average distance between the predicted and actual location

It can be seen from Table 8 when W is -86 dBm and X is 5 an average distance of 192.51 inches, which is the minimum value, is obtained. Hence this is the condition with the best result. This is also consistent with the results we

Table 8: Analysis of average distance between the predicted and actual location

Weak signal Threshold (W) (dBm)	SS Range Window size (X) (dBm)	Average Distance to the Actual Location (inches)
-80	12 (± 6)	242.47
-80	10 (± 5)	224.5
-83	12 (± 6)	210.77
-83	10 (± 5)	193.56
-86	12 (± 6)	214.43
-86	10 (± 5)	192.51

obtained in the previous analysis.

5.3.3 Performance of Corridor vs. Other Rooms

Upon analyzing the physical location of the APs, it was determined that one of the seven APs was located in the first floor corridor. Hence it was hypothesized that if an analysis of performance of corridor vs. other rooms were done, the prediction will be better in the corridor. This was initially tested with a set of 28 locations and the as expected, the prediction at corridor was better. In including test results from 42 more locations, Figure 5 was obtained as the result of the analysis of the 70 points together. The prediction in the rooms appears comparatively better than the corridor. The reason for this has not been determined. We still believe there must be more testing done and the corridors will eventually perform better.

5.4 Role of Orientation in Accuracy of Predicting Correct Rooms

To determine the effect of the orientation on prediction of location, for each location the SS values at different directions were averaged and a new radio map was created. The prediction of locations for the test data was done on this radio map. It was observed that for the best case of $W = -86$ and $X = 5$ there was approximately 70% accuracy in prediction of correct room while with orientation as mentioned before, for the same parameter values of $W = -86$ and $X = 5$, there was an accuracy of 77%. Hence it is deduced that direction did play a considerable role in accuracy of prediction.

5.5 Visualization of Predicted Points

To understand the result of prediction, the results were visualized by plotting all pairs of an actual location and a predicted location on the floor map of Lindley Hall. Figure 6 depicts the result of the predictions. From Figure 6, it can be

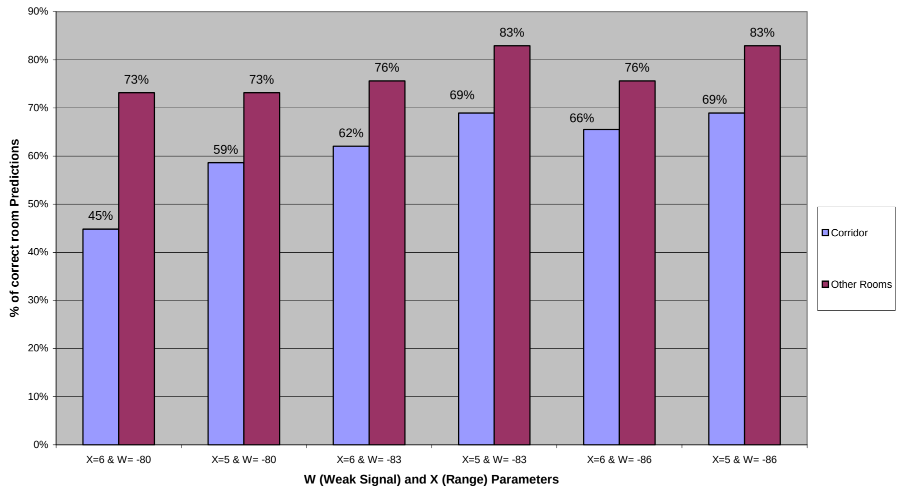

Figure 5: Performance of corridor Vs rooms

deduced that for most of the pairs, the prediction is in the same room. Even when the predicted points are in other rooms, the rooms were observed to be adjacent. However, it should be noted the prediction of some points in room LH115 are located in room LH101 or LH102 which is considerably remote. It is speculated the reason for these mis-predictions could have been due to the characteristics of the radio map of LH115.

According to the radio map (Figure 1), there is only one strong signal (lindley_hall_104-1) in LH115. Because the prediction algorithm doesn't count weak signals, it is hard to predict correctly only with one strong signal. As a result, it predicts points in LH101 or LH102 that also have the strongest signal from lindley_hall_104-1.

6 Prototype System

6.1 Overview

The prototype system implemented is an application that shows a user's location in Lindley Hall. A user simply clicks a button to see where she is on the map generated.

Throughout the evaluation phase, the entire system was on the MS, including the database of radio map and the prediction computing. Since the goal of this project is to deploy a client to each laptop and PDA in Lindley Hall, this model is not practical. PDAs, especially, do not have enough computation power, or memory space to store the radio map. The prototype system is designed based on the client-server model, moving computation and data to the server side.

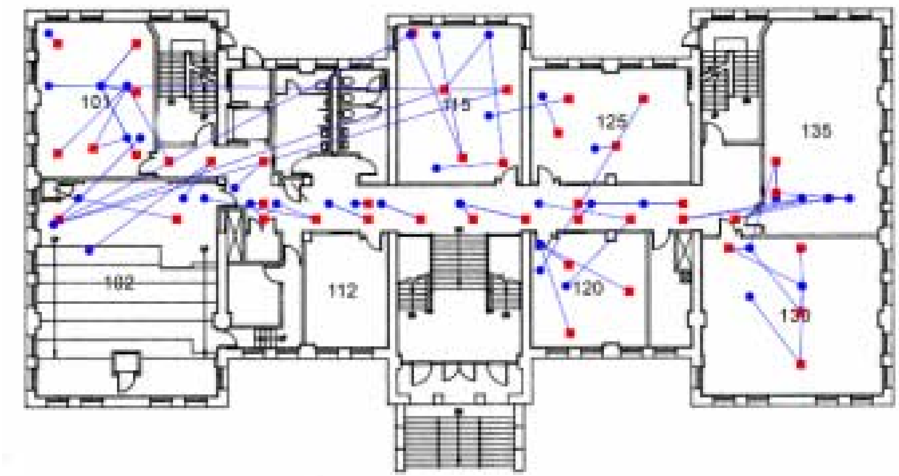

Figure 6: A map with predicted points plotted. Pairs of points connected represents pairs of actual location and predicted location. Rectangular points indicate actual locations, and circular points predicted locations.

The function of client is simply acquiring pairs of MAC addresses and SSs, sending them to the server, and displaying a map created on the server to the user. Likewise, the server receives the MAC/SS pairs from a client, predicts the user's location, generates the map, and sends this information back to the client.

6.2 System Architecture

To make the system compatible with clients on various devices, standard HTTP Get message is employed for client-server communication. By following standard HTTP protocol, which is supported by most PDAs, even by some phones, it is estimated to be easier to implement clients on various small devices in the future.

Figure 7 depicts the system architecture and the execution flow. A client is a Windows application implemented in Visual Studio .Net C#. When a user click a “Locate” button, it acquire pairs of MAC addresses and SSs from the wireless card of the client computer, and generates the URL by combining the base URL address with the acquired pairs. Then, it sends the URL as a standard HTTP Get message by TCP. In other words, it opens the page indicated by URL in the same way as a standard web browser does. After the server creating the image of a map with the predicted point, the client displays it to the user.

The front-end server is a standard web server (Apache 1.3.29), with a radio-map database back-end (MySQL 4.0.16). A CGI written in Perl receives a

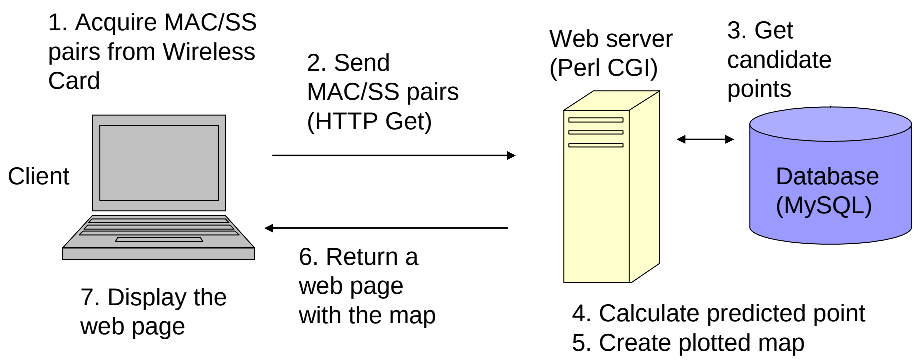

Figure 7: The architecture of the prototype system

request from a client, and as described before, makes a query to obtain candidates points from the radio-map database. From the candidates point, using Euclidean distance, it decides a final prediction point. With the help of an image manipulation toolkit (ImageMagick 5.5.7), the CGI produces a image of a map of Lindley Hall with the predicted point. Finally, it returns a web page with the created map to the client.

7 Comparison with Other Systems

RADAR has the advantage that it requires only a few base stations, and it uses the same infrastructure that provides the building's general purpose wireless networking. RADAR suffers from the disadvantage that the object it is tracking must support a wireless LAN, which may be impractical on small or power-constrained devices and generalizing RADAR to multi-floored buildings is nontrivial.

Implementation of RADAR done for the department building can track users within 192 inches of their actual position within a 90 percent probability.

The Bat system can locate Bats to within 46cm of their true position for 96.7 percent of the measurements and work to improve the accuracy even further is in progress. It can also compute orientation information given predefined knowledge about the placement of Bats on the rigid form of an object and allowing for the ease with which ultrasound is obstructed. Each Bat has a GUID for addressing and recognition. Using ultrasound time of flight this way requires a large fixed-sensor infrastructure throughout the ceiling and is rather sensitive to the precise placement of these sensors. Thus, scalability, ease of deployment, and cost are disadvantages of this approach.

Unlike the Active Bat system Cricket does not require a grid of ceiling sensors with fixed locations because its mobile receivers perform the timing and computation functions. Cricket, in its currently implemented form, is much less precise than Active Bat in that it can accurately delineate 4x4 square-foot regions within a room, while Active Bat is accurate to 9cm. However, the fundamental limit of range-estimation accuracy used in Cricket should be no different than Active Bat, and future implementations may compete with each other on accuracy.

Cricket has the advantages of user privacy and decentralized scalability but it has the disadvantage of lack of centralized management or monitoring. Cricket also suffers from the problem of battery discharge of beacons and listeners. Typically battery life of cricket is 2 weeks if used in continuous operation.

Table 9 is the tabular comparison of three systems in terms of cost infrastructure and setup time.

Table 9: Comparison of three systems

Criteria	RADAR	Cricket	Bat
Accuracy	3-4m (50%)	4x4 ft region (~100%)	46 cm (96.7%)
Cost Infrastructure	Wireless Lan	$12,500 Beacons	$250,000
Cost Devices	$30 for NIC	$10 per Listener	$100 for Bat
Setup Time	Less	High	High

8 Future Work

Due to time constraints of the project period there are still a number of areas for further study.

First, it is important to evaluate how different wireless LAN cards affect measurement of SS and prediction. It is not reasonable to assume that all users use the same wireless LAN card in order to deploy the system widely. There will be various types of wireless cards used, such as a regular PCMCIA type, an internal type, or a compact flash type for PDAs. While the prototype system was implemented, a different kind of laptop was used, and the prediction was not affected by difference of wireless cards. For example, even with a laptop with an internal wireless LAN card, the prototype worked fine. However, in order to conclude this, it is necessary to formally measure SS with different wireless LAN cards at the same time and at the same location, and to compare the result.

Second, more effort to enhance the accuracy should be made. There are various sophisticated prediction algorithms to try, such as machine learning or improvement of accuracy can be made by utilizing user's previous location (continual user tracking). The result shows that predictions are correct most of the time, but once in a while it predicts a location far from the actual location. By using the user's previous position, it is possible to prevent this jumping. The prediction algorithms can be enhanced by putting more weight on a locations closer to the user's previous position. To accomplish this, a client needs to send a unique ID, such as MAC address of its wireless card, to the server along with pairs of MAC address and SS. Then the server keeps each user's past locations in the database, and use it to predict future location. However, there are several issues for this approach. First of all, a client needs to send information to the server periodically, which causes a severe energy consumption problem. Also, users' privacy will be an issue to consider. Most importantly, we need to evaluate what kind of algorithms are suitable for this approach.

Last of all, mapping the rest of Lindley Hall is necessary in order to observe how the prediction works in 3D-space. There are some new issues expected by extending the system to the entire Lindley Hall. First of all, scalability might become an issue. Even-though in the current system, it does not take much time to predict a location, it is not certain how the approach scales when the data becomes five-times bigger. Secondly, it is not known how accurate the system can differentiate Z-coordinate. For example, it important to know if the system can predict LH130 as LH130, not as LH230. To prevent that a user suddenly jumps from 2nd floor to 1st floor, it might be important to utilize user's previous location as described above.

A Bat Ultrasonic Location system

A.1 Overview

The Bat Ultrasonic Location System[2] was developed by AT&T Laboratories Cambridge, It succeeds their preceding work "Active Badges[16]."

Different from Active Badges, the Bat system uses a combination of the radio link and ultrasound.

In the Bat system, a small device called "bat" is attached to objects or carried by people. It emits a short pulse of ultrasound, triggered periodically by a central controller called a base-station. The pulse is received by receivers mounted on the ceiling. The system measures the times-of-flight of the pulse, and calculates the distance from the bat to each receiver. Given three or more such distances, the system can determine the 3D-position of the bat. By attaching two or more bats to an object, it is also possible to calculate its orientation.

A.2 Components

Bat A bat is a small device attached to an object or carried by people. The size is 7.5cm $\times$ 3.5cm $\times$ 1.5cm. It is powered by a single 3.6V lithium thionyl chloride cell, which lasts approximately fifteen months. It has an ultrasound emitter; a unique 48-bit code; a bidirectional 433MHz radio link; two input buttons; two LEDs; and a voltage monitor for battery status.

Receiver The receivers are attached to ceilings to detect the ultrasonic signals from bats. Receivers are placed 1.2m apart from each other. They are connected by a high-speed serial network in a daisy-chain fashion. The serial network is terminated by a DSP calculation board. The DSP calculation collects results from the receivers and calculate the positions of the bats.

Base station A central controller, also called a base-station, coordinates the bats and the receivers. It periodically transmits a radio message containing a single identifier, causing the corresponding bat to emit a short pulse of ultrasound.

A.3 Software

The software component of the Bat system is implemented using CORBA, Oracle, and Ouija. The physical entities are expressed as objects of CORBA, and stored in an Oracle database. Ouija is used as object-oriented data modeling language, to generate an object layer on top of the Oracle database.

The system employs some proxies which are responsible to convert CORBA calls to a PL/SQL operation, and also offer a fast path for information that is updated too frequently to store in the database.

A.4 Specification for Lindley Hall

In this section, we evaluate the application of the Bat System to Lindley Hall, and make a specification.

A.4.1 Bats

There are 20 faculty, 18 staff, and 424 graduate and undergraduate students in the Indiana University Computer Science Department (462 people total). Assuming only those who have offices (171) occupy Lindley Hall frequently, the initial number of bats required is 200 including spares.

Unfortunately, I was not able to get price information of a bat from the developer of the Bat system. According to a commercial company, Hexamite[1], which sells a similar ultrasonic location system, their ultrasound receiver[5] costs $369 USD. I thought this was over-priced, but ultrasonic related products cost almost the same everywhere.

The cheapest ultrasonic receiver hardware device I have found[6] is by the same company, and costs $49 USD. We estimate by assembling from this hardware, we can reduce the price of each bat to less than $100 USD.

Here, we approximate each bat to cost $100 USD, and estimate the total cost of 200 bats to be $20,000 USD.

[5] HE900T Hexamite Positioning Device with I/O pins and Internal Rechargeable Battery (HE900T-PQ)

[6] HE225TXB_Transmitter

A.4.2 Receivers

Receivers are placed 1.2m apart from each other. That means one receiver for each $1.44\mathtt{m}^2$ (= 15.5 squire-feet). Since there are so many receivers in each room, we only estimate the number of receivers in each room, instead of plotting actual positions on the map.

Table 10: Number of receivers in each room

Room	Area (squire-feet)	Number of receivers
LH101	652.5	42
LH102	1377.0	89
LH115	724.5	47
LH125	645.0	42
LH135	880.0	57
LH130	958.8	62
LH128	273.0	18
LH120	451.5	29
LH112	325.5	21
Corridor	1346.0	87
Total	7633.8	494

Table 10 shows the number of receivers needed for each room. To cover the whole first floor of Lindley Hall, we need 494 receivers.

Again, the price of a similar device[7] form Hexamite is $369 USD. By assembling the receiver from the cheapest hardware I found on the web[8], which costs $49 USD, we can reduce the price of each receiver to less than $100 USD.

Here we assume the price of receiver as $100 USD. Then, the cost for the first floor of Lindley Hall would be approximately $50,000 USD, and for the entire five floors it would be approximately $250,000 USD.

A.4.3 Base Stations

According to the paper, the authors used two base stations to cover 10,000 square-feet. So, it is reasonable to use three base stations for the first floor of Lindley Hall. One is placed in LH102, one in LH115, and the other in LH130, to separate them from each other.

RF emitters are generally cheaper than ultrasonic devices, each base-station costs around $50 USD. The cost for the first floor of Lindley Hall would be approximately $150 USD, and for the whole 5 floors it would be approximately $750 USD.

[7] HE900M1 Hexamite Positioning Device Serial I/O Option (HE900M1-Serial)

[8] HE225RXB_Receiver

Table 11: Length of wire in each room

Room	Number of receivers	length (feet)
LH101	42	171
LH102	89	360
LH115	47	195
LH125	42	175
LH135	57	234
LH130	62	254
LH128	18	81
LH120	29	124
LH112	21	93
Corridor	87	352
Total	494	2039

A.4.4 Wiring

Receivers attached to ceilings are placed 1.2m apart from each other, and connected by serial cables in a daisy-chain fashion.

Table 11 shows the length of wire needed for each room of the first floor of Lindley Hall. It includes wire to connect each of the receivers and an extra 10 feet to connect to other rooms. The total length of wire to cover the first floor is 2039 feet.

We estimate $50 USD for 100-feet serial cable. Thus, the cost of wiring for the first would be approximately $1,000 USD, and for the entire five floors it would be approximately $5,000 USD.

A.4.5 Total Cost

Table 12 shows the summary of the cost. The total cost to equip the Bat system for the entire Lindley Hall is estimated at approximately $280,000 USD.

Table 12: Total cost

	1st floor (USD)	Entire Lindley Hall (USD)
Bats	—	20,000
Receivers	50,000	250,000
Base stations	150	750
Wiring	1,000	5,000
Total	—	275,750

A.5 Granularity

According to their most up-to-date paper, approximately 95 percent of 3D bat positions have accuracy within 3cm.

They installed the Bat system throughout a three-floor, 10,000 square-feet building. They used 750 receivers and 3 base stations, tracking 200 bats. Each base-station address 3 bats simultaneously, 50 times each second, giving a maximum 150 location updates per second.

A.6 Discussion, and Summary

The Bat system gives us fine granularity of location, also orientation of objects and people. These features are very useful for certain applications. However, merely focusing on the application to the scenario described in [10], the system is over specification and the cost of set-up is high.

To lessen the cost, we might be able to use less receivers, but analysis concerning the density of receivers has not been performed.

Helal et al.[12] has developed another ultrasonic-based location system using devices from Hexamite. The most noticeable difference between this system and the Bat system is that it uses ultrasound in both ways; on the other hand, the Bat system uses both RF and ultrasound. This system equips ultrasonic pilots only at each corner of a 500-square-foot house, and 96.7 percent of the position predictions are within 40cm. This system is more cost effective.

The developers of the Bat system are now developing a commercial location system using ultrawideband (UWB) radio[4]. The accuracy is around 15cm in 3D space. However, this system requires less infrastructure resulting in better price-performance.

B Cricket Location System

B.1 Overview

Cricket[14] is an indoor location system for pervasive computing environments. These environments take advantage of emerging network-enabled devices and the promise of ubiquitous network connectivity.

A compelling set of applications in pervasive environments are context-aware, being able to discover the external context in which they run and adapt accordingly. An important example of context is location, such as the position (in some coordinate system) of a device or user, the geographic space in which a device or user is (e.g., the room or portion of a room), and the orientation of a device within some coordinate system.

Knowledge of location in the form of coordinate position, spatial resolution, and orientation enables a wide variety of pervasive computing applications such as resource discovery, "point-and-use" interfaces, navigation, and augmented reality.

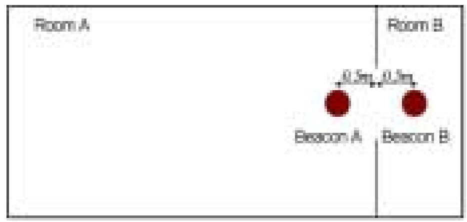

Figure 8: Beacon's placement

Obtaining location information for applications in an indoor environment in an unobtrusive and private manner is a challenging task. Indoor environments are harsher than outdoor ones in their treatment of radio signals because of multi-path effects and dead spots inside buildings. A traditional magnetic compass doesn't work well in many buildings with computers and monitors because of EM interference. User-privacy concerns are an important consideration in the successful deployment of these applications, especially if the users of the system are to extend beyond the researchers who develop the technology. The administration of the hardware and software infrastructure used for this must be minimal because of the large number (potentially over several thousand in a typical building) of devices and networked services that need this information.

B.2 Hardware

The Cricket hardware[3] consists of the beacon and listener. They both take 2 AA batteries. NiMH rechargeables are better in terms of their capacity. Using a set of fully charged NiMH batteries, the beacons last about one to two weeks in continuous operation.

The Cricket beacons are the larger modules with the red wire antenna. A typical beacon has a transmission range of about 10m (depends on the environment; one can also use various simple hardware hacks to increase/decrease this range). Each beacon is pre-programmed to broadcast a unique space identifier, which is used by the listener to identify the user's location. Hence, there should be at least one beacon in each space (e.g. a room). The correct beacon placement is as shown in Figure 8.

The beacons can also be used to define two or more spaces within the same room. In this case, a pair of beacons can be placed at about 1-1.5m apart. This effectively defines a virtual boundary that divides the spaces represented by each beacon.

B.3 Software

All Cricket software currently runs on both Linux and Windows platforms that runs JDK1.3 or above.

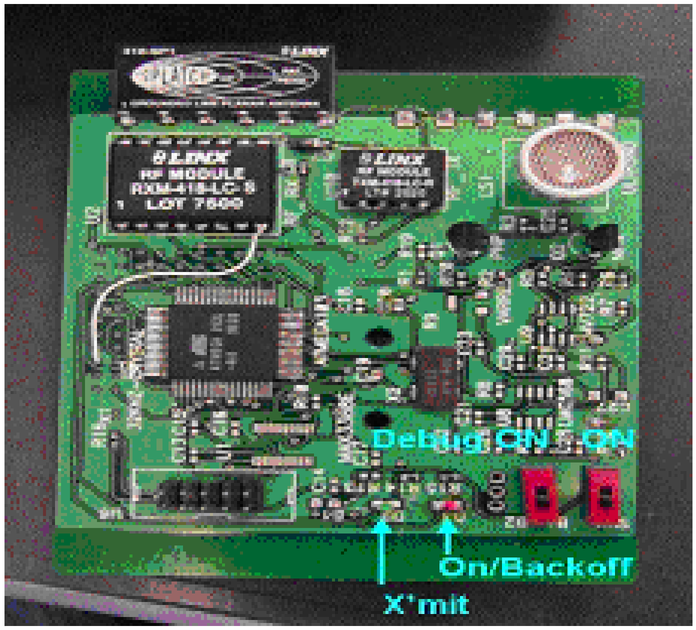

Figure 9: Figure of Cricket Beacon

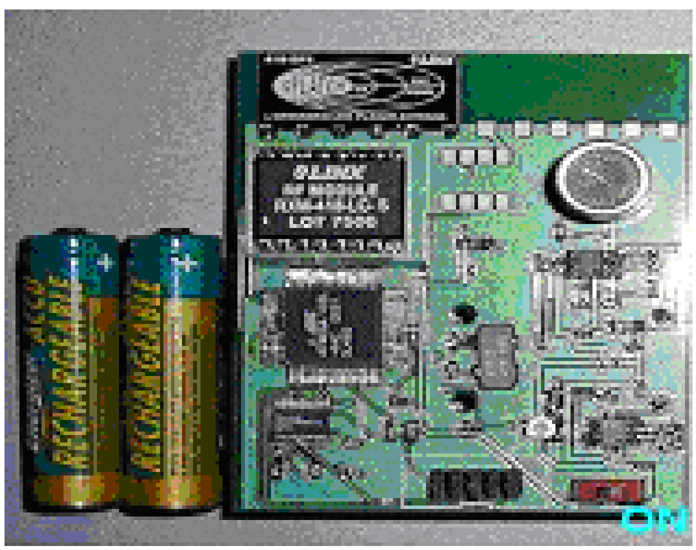

Figure 10: Figure of Cricket Listener

The software consists of applications and a daemon, which processes readings from the Cricket listener, attached to the serial port and exports the processed reading to the applications.

B.4 Technology

Cricket uses a combination of RF and ultrasound technologies to provide a location-support service to users and applications. Wall- and ceiling-mounted beacons are spread through the building, publishing information on an RF signal operating in the 418 MHz AM band. With each RF advertisement, the beacon transmits a concurrent ultrasonic pulse. Listeners attached to devices and mobiles listen for RF signals, and upon receipt of the first few bits, listen for the corresponding ultrasonic pulse. When this pulse arrives, they obtain a distance estimate for the corresponding beacon. The listeners run maximum-likelihood estimators to correlate RF and ultrasound samples (the latter are simple pulses with no data encoded on them) and to pick the best one.

B.5 Results

Cricket uses active beacons and passive listeners, which has two significant benefits. First, it is not a tracking system where a centralized controller or database receives transmissions from users and devices and tracks them. Second, it scales well as the number of devices increases; a system with active transmitters attached to devices wouldn't scale particularly well with the density of instrumented devices. Third, its decentralized architecture makes it easy to deploy. This does not mean it is hard to manage; a centralized front-end allows easy management and control.

Cricket can determine which space a device is in by detecting boundaries to within about 2 feet. Beacons are placed in 5 x 5 feet grid. So it can give location granularity of 5 x 5 feet and it can determine angles to within 3-5 degrees of the true value. Beacons are placed at the ceiling where they have better line of sight which gives more accuracy.

B.6 Costs

Estimated Cost of Using Cricket in Department Building based on estimated cost of placing beacons on each floor.

A listener need to be attached to every mobile device that's need to be tracked. Every listener cost $10. Total cost will depend upon number of mobile devices within the department building.

References

[1] Hexamite. http://www.hexamite.com/.

Room	No. of Beacons	Cost/Beacon	Total Cost
Ground Floor	250	$10	$2500
First Floor	250	$10	$2500
Second Floor	250	$10	$2500
Third Floor	250	$10	$2500
Fourth Floor	250	$10	$2500
			$12,500

[2] The Bat Ultrasonic Location System. http://www.uk.research.att.com/bat/.

[3] The Cricket Indoor Location System. http://nms.lcs.mit.edu/projects/cricket/.

[4] Ubisense. http://www.ubisense.net/.

[5] Wireless Research API (WRAPI). http://ramp.ucsd.edu/pawn/wrapi/.

[6] Wireless Tools for Linux. http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html.

[7] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementing a Sentient Computing System. *IEEE Computer Magazine*, 34:50–56, August 2001.

[8] Paramvir Bahl and Venkata N. Padmanabham. Enhancements to the RADAR User Location and Tracking System, February 2000.

[9] Paramvir Bahl and Venkata N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proceedings of Infocom 2000*, pages 775–784, March 2000.

[10] Kay Connelly and Ashraf Khalil. Towards Automatic Device Configuration in Smart Environments. In *Proceedings of the Fifth Annual Conference on Ubiquitous Computing (UbiComp 2003)*, October 2003.

[11] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The Anatomy of a Context-Aware Application. In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM'99)*, pages 59–68, September 1999.

[12] Sumi Helal, Bryon Winkler, Choonhwa Lee, Youssef Kaddourah, Lisa Ran, Carlos Giraldo, and William Mann. Enabling Location-Aware Pervasive Computing Applications for the Elderly. In *Proceedings of the First IEEE Pervasive Computing Conference (PerCom 2003)*, June 2003.

[13] Ashraf Khalil. Deploying RADAR system in Lindley Hall: A Project Proposal.

[14] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The Cricket Location–Support System. In *Proceedings of the Sixth Annual ACM International Conference on Mobile Computing and Networking (MOBICOM)*, August 2000. http://nms.lcs.mit.edu/papers/cricket.html.

[15] Pete Steggles, Paul Webster, and Andy Harter. The Implementation of a Distributed Framework to support 'Follow Me' Application. In *Proceeding of the 1998 International Conference on Parallel and Distributed Processing Technique and Applications (PDPTA'98)*, volume 3, pages 1381–1388, July 1998.

[16] Andy Ward, Alan Jones, and Andy Hopper. A New Location Technique for the Active Office. *IEEE Personal Communications*, 4(5):42–47, October 1997.