Indoor Wireless Local Area Network (WLAN)

Measurement and Modeling from a user perspective

B E H D I S   Z A N D I E H

Indoor WLAN

Measurement and Modeling from a user perspective

Behdis Zandieh

A thesis submitted in partial fulfillment of the requirements for the
Degree of Master of Science in Computer Engineering

28 March 2007

School of Information and Communication Technology

Royal Institute of Technology (KTH)

Stockholm, Sweden

Examiner: Professor Gerald .Q. Maguire Jr.

Abstract

If we had detailed wireless local area network (WLAN) coverage maps, both staff and equipment could be used more efficiently, for example, less time would be spent searching for connectivity. In addition, system administrators could understand their WLAN's utilization better, thus enabling better planning for where to install new access points, where to remove access points, where to change the type of antenna, etc.

This thesis concerns creating detailed indoor coverage models by using measured network performance - in order to enable both users and administrators to visualize the network coverage. Today a user can only easily know about the access points that they currently hear – in the location where they currently are. Giving the users and administrators access to a model of the entire campus coverage will allow them to understand not only the _local_ coverage, but the patterns of coverage (or lack there of). However, no efficient modeling techniques are currently available for those deploying and operating indoor WLANs.

The thesis begins with some general background information and then examines a number of WLAN survey tools; in terms of both their performance and cost. Following this a number of related projects are presented. This background provides the motivation for why a new tool is needed and what functions such a tool should have.

Next a site survey of the KTH campus in Kista was conducted using a newly developed survey application. This application was developed to better meet the requirements derived from the missing functionality of existing tools. In addition, developing this application gave the author an opportunity to learn a new object-oriented programming language, i.e. C# and the .NET environment. Learning to use this new environment was essential to building both an easy to use application and collecting the data from the system - the later was often not straight-forward.

One of the key issues after data collection is how to present the collected data to the user and how this varies depending upon the user's interests and task. An initial representation of the experimental data is presented as a manually painted coverage map overlayed onto a map. Next the thesis examines how to integrate the experimental data using Google's SketchUp in order to build a 3D model of the WLAN coverage on this campus.

Future work related to this thesis should focus on how to automate the collection of data and how to automate the presentation of the resulting experimental data.

Sammanfattning

Om vi hade tillgång till detaljerade täckningskartor över lokala nätverk (WLAN), skulle både personal och utrustning kunna användas mer ändamålsenligt. Till exempel skulle mindre tid gå åt till att söka täckning. Dessutom skulle systemadministratörer förstå användningen av deras WLAN:s på ett bättre sätt, vilket skulle möjliggöra bättre planering av var nya accesspunkter ska installeras, var accesspunkter ska tas bort, var man ska byta antenntyp, osv.

Detta examensarbete handlar om skapande av detaljerade täckningsmodeller för användning inomhus vilka genererats genom att mäta nätverkets styrka – detta för att göra det möjligt för båda användare och administratörer att visualisera nätverkstäckning. Idag kan en användare endast känna till accesspunkter som de just för tillfället hör – på den plats där de för tillfället befinner sig. Att ge användarna och administratörerna tillgång till en modell av täckningen över hela Campus skulle inte bara göra att de uppfattade den lokala täckningen utan hela täckningsmönster (eller avsaknaden därav). Dock finns för tillfället inga ändamålsenliga modelleringstekniker för de som utvecklar och sköter WLAN:s inomhus.

Examensarbetet inleds med en del bakgrundsinformation och går därefter in på ett antal metoder för att kartlägga WLAN, både vad gäller prestanda och kostnad. Härefter presenteras ett antal relaterade projekt. Denna bakgrund är till för att motivera varför ett nytt verktyg behövs och vilka funktion ett sådant verktyg borde ha.

Efter denna litteratur- och bakgrundsstudie gjordes en kartläggning av KTH Campus i Kista med användning av en nyligen utvecklad applikation. Applikationen togs fram för att på ett bättre sätt uppfylla kraven som följde av saknade funktioner i existerande verktyg. Dessutom gav utvecklingen av applikationen författaren en möjlighet att lära sig objektorienterad programmering med t ex C# och .NET-miljön. Att lära sig att använda denna nya programmeringsmiljö var en nödvändighet för att kunna bygga både en lättanvänd applikation och samla in data från systemet – de senare var oftast inte helt lätt.

En av de största svårigheterna efter datainsamlingen var att hitta ett sätta att presentera den insamlade informationen samt hur presentationen ska variera beroende på användaren behov och uppgift. En första presentation av experimentdata presenteras som en manuellt ritad täckningskarta lagd ovanpå en vanlig karta. Nästa steg i examensarbetet är att undersöka hur man ska integrera experimentdata genom att använda Google:s SketchUp för att bygga en 3D-modell av WLAN-täckning på Campus i Kista.

Framtida utveckling relaterad till detta examensarbete borde fokusera på hur man ska kunna automatisera datainsamlingen och presentationen av resulterande experimentdata.

Dedication

To my mother, Dr. Fariba Mozaffari.

Acknowledgements

I would like to express my sincere gratitude to my advisor Professor Gerald Q. Maguire Jr. for the opportunity to work in his lab and for his continual help and guidance on this project. His ideas and advice have been decisive in my efforts to carry out this thesis.

I also would like to thank the personnel in the Wireless@KTH research center for welcoming me and helping me whenever I have needed during my stay.

Table of Contents

List of Figures

List of Tables

1. Background

1.1. Wireless LAN

The difference between wireless LAN (WLAN) (also called WiFi (Wireless Fidelity) or IEEE 802.11 networking) and a wired LAN is that a WLAN does not use cables as its transmition media. Rather electromagnetic waves, radio waves or infrared light are used. This avoids the cost of installing cables and enables access to the media from non-fixed locations.

Most WLANs consist of two parts: end-user devices and Access Points (AP). Both of these elements are equipped with a wireless interface that translates data into a radio/IR signal and transmits it using an antenna or emitter. Each AP functions like as a bridge after receiving the signal and decoding it. The AP provides synchronization and coordination of devices within a local area, forwarding of packets, and perhaps bridging packets to and from a wired network.

Every access point is distinguished from other access points by a unique hardware address called a Media Access Control address. Every access point has a wireless interface that understands the WLAN's media and access control protocol and may have an interface connected to the wired LAN. The wired interface is almost always an Ethernet port.

Access points could even have multiple wireless interfaces to increase the hot-spot capacity by using two interfaces on nonoverlapping channels. Some access point could use an external antenna to be able to send the signals further or in a specific direction.

There is a buffer memory that holds the frames while they are transferred between the two interfaces, and MAC address of each port is stored in a set of internal tables. These tables are updated based upon the normal spanning tree algorithm used in bridges.

The basic structure of a WLAN is called a Basic Service Set (BSS), which comes in two flavors: Independent BSSs and infrastructure BSSs. In an independent BSSs (IBSS) stations communicate directly with each other and are usually composed of a small number of stations set up for a specific purpose, often only for a short period of time. IBSSs are often referred to as *ad hoc* networks. See Figure 1 .

Infrastructure BSS consists of an AP and several wireless devices. These devices communicate with each other through the AP. See Figure 2.

The AP broadcasts its Service Set Identifier (SSID) to allow others to join this specific network. Each BSS is assigned a BSSID, this is a 48-bit binary identifier that distinguishes it from other BSSs. In infrastructure BSSs, the BSSID is the MAC address of the wireless interface in the access point creating the BSS. Ad hoc networks generate a random BSSID.

| Figure 1. Independent Basic Service Set | Figure 2. An infrastructure Basic Service Set |
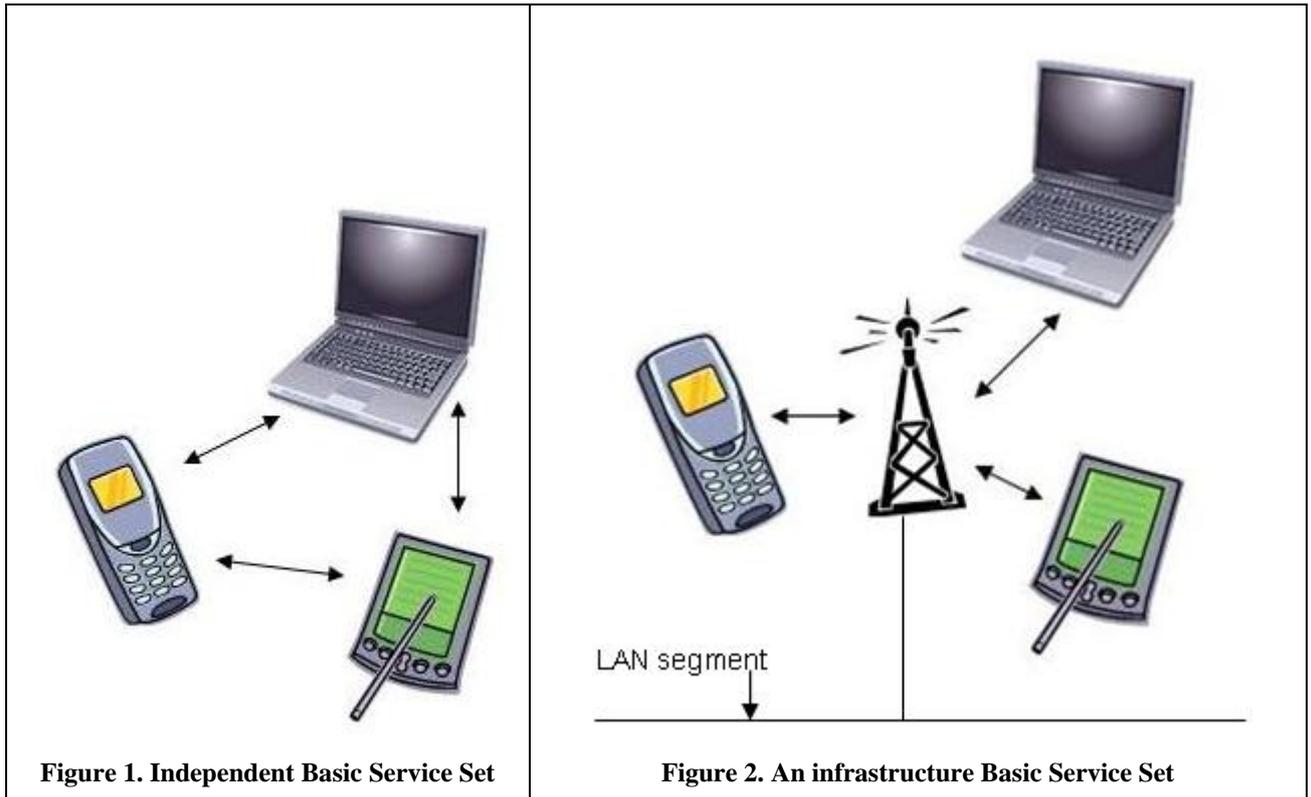
The end-user devices always initiate the association process, while access points choose to grant or deny access based on the contents of this association request.

As users move, their mobile device will attempt to find and connect with the BSS providing the strongest signal and the least amount of competing traffic.

The radios used for WiFi communication are very similar to the radios used for phones, but they differ with respect to the frequencies used, for example IEEE 802.11 b at 2.4 GHz or IEEE 802.11a at 5GHz (which are both higher frequencies than used by cellular phones) additionally they use much wider bandwidths which allows the signal to carry more data per unit time.

The IEEE 802.11 networking standards comes in several flavors. IEEE 802.11 [1] is the oldest, and only supports 1 or 2 Mbps. It operates in the 2.4 GHz Instrumentation, Scientific, and Medical band. Operation in this band allowed unlicensed users to use approved equipment but did not require the users to have a license- which enabled a broad user base. IEEE 802.11b supports up to 11 megabits of data per second. 802.11g is a further enhancement to support up to 54 megabits of data per second. 802.11g is faster than 802.11b because it uses a more efficient coding technique.

IEEE 802.11a operates in the 5 GHz band and supports up to 54 megabits/second.

1.2. Cellular communication

1.2.1. General system architecture

Because of the limited radio spectrum assigned, cellular networks exploit the finite range of signal propagation, thus a given frequency can be reused by cells that are sufficiently far away- so as to avoid interference.

Cells are generally modeled by hexagons and base stations are located in the centers of them. Each base station has a certain range associated with it. To avoid radio interference, these channels must be different from the channels associated with all of its neighboring cells. However, in the case of IEEE 802.11b, in high speed mode this is not possible, so there will be some unavoidable interference -- as there are only three non-overlapping frequency allocations than can be used -- where as 4 would be required to cover a planar area.

A group of N adjacent cells in which all radio channels are used is called a cluster. Figure 3 shows a simplified cellular architecture.
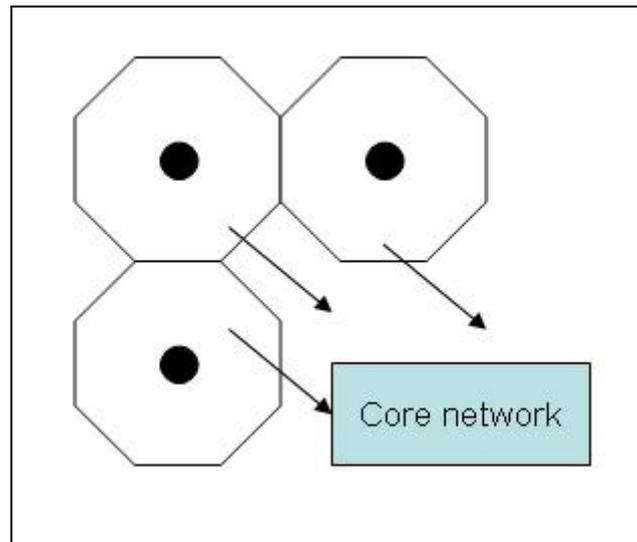
Figure 3. Cellular network architecture

The specific functions implemented by the components of the cellular networks can vary from standard to standard and vendor to vendor, but I will briefly give a general overview of such systems.

The area covered by a cellular network is divided into areas called cells (In some networks cells are further divided to sectors - using directional antennas) and each cell has a base station (BS) which communicates with all the mobile stations within the cell, it also forwards and passes the traffic from these mobile stations to the core network or distribution system.

In the case of WLAN, each mobile station constantly monitors the received signal strength as a user moves from one location to another location the strength of the signal it receives from the access point which it is currently associated will change. When a mobile station is near the border of a cell the signal will be weak. At a certain point the mobile station will look for stronger signals from other access points. The strongest received signal from an access point (perhaps one in another cell) is the one that the mobile will pick, and this procedure is called *handover* procedure, i.e., this mobile station will ends it association with its current access point and try to associate with the new access point.

As we can see, where handover is likely to take place is the point where the neighbor's signal is strong than the current access point's signal. Hence understanding the signal strength of each access point as a function of position can describe where a mobile station will get coverage and where it is likely to want to perform a handover.

Determining the signal strengths of the set of access points as a function of position is called WLAN surveying and it is the topic of the next section.

1.3. WLAN surveying

1.3.1. What WLAN surveying means

The size of the WLAN market is growing rapidly as the flexibility, connectivity, mobility, and low cost of this technology meets the needs of many consumers. Moreover, utilizing coverage maps can help users to find places with connectivity. Creating these coverage maps requires WLAN surveying tools. The measurement data from such surveys can also be used to create a coverage model, which goes beyond simply mapping the collected data.

A simple way to find wireless networks is to perform a site survey with a computer equipped with a wireless network card. One of the earliest methods to collecting such data about a large area is wardriving.

Wardriving involves people using a computer equipped with both a wireless card and a GPS receiver, and then driving around a city looking for the presence of wireless networks [2]. By using a GPS receiver wardrivers can record the exact location of their measurements and hence locate an access point in world coordinates. Besides wardriving there is also a term for people who perform site surveying via walking: warwalking.

A key result of wardriving is the list of access points and locations, along with additional information such as available data rates, signal strength, and whether it is an open or closed network, i.e., if they allow others to use their network. An example of this data is shown in Table 1. This data was collected using the program, WiFiFoFum [3] (a WiFi scanner program) in the computer communication systems lab in the Wireless@KTH center.

Table 1. Sample data collected by WiFiFoFum

MAC	SSID	RSSI	Channel	Time
00022DB43658	Default	-41	10	1/17/07
00022D920161	Open	-50	1	1/18/07
000CE6882820	Vowlan	-87	6	1/18/07
00022D9201A0	Open	-78	11	1/18/07

1.3.2. Useful measurement tools

There are many shareware and commercial tools available to help measure WLAN signal strength and related metrics. These tools scan all the channels and record information describing every discovered AP. Generally, these programs have come to be referred to as "stumblers" after one of the early programs called "NetStumbler" [4]. Most stumblers can display AP signal strength, noise level and signal to noise ratio (SNR), some of them can record this information to a file for later use.

As the focus of this work is going to be a campus we will utilize tools which can run on a Pocket PC - to facilitate data collection by a user walking around this campus. Thus here we will focus on tools for this platform.

Figure 4 shows the WLAN coverage of the 4rth Floor of KTH's Forum building and is an example of how the measured signal strength values could be used to build a coverage model as you see different colors indicate different levels of signal qualities.
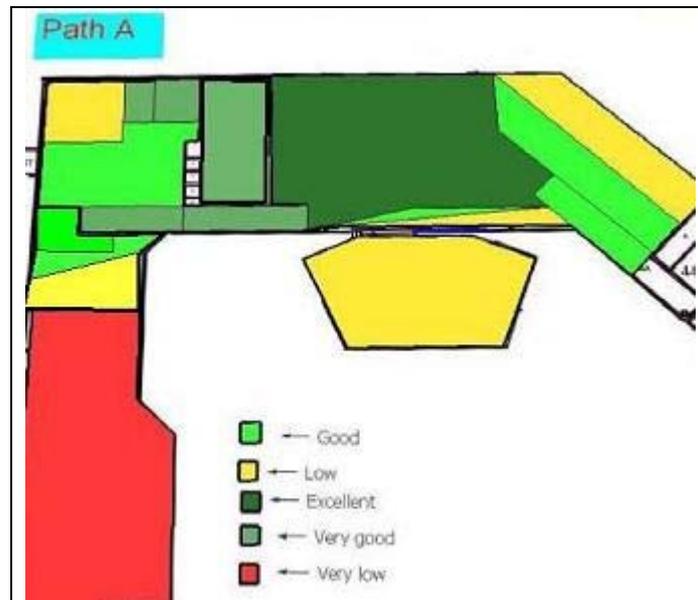
Figure 4. WLAN coverage example map

Several freely available survey tools for Pocket PCs are:

1. MiniStumbler is the original wireless LAN scanning utility for Windows. It will work with HPC2000, PocketPC 3.0, PocketPC 2002, and Windows Mobile 2003 [5].
2. WiFiFoFum is a WiFi scanner designed for PDAs running PocketPC 2003 and Windows Mobile 2005. See Figure 6.
3. Pocket Warrior is wireless auditing software for PRISM and NDIS 5.1 compatible card that runs on PocketPC 2002 [6].
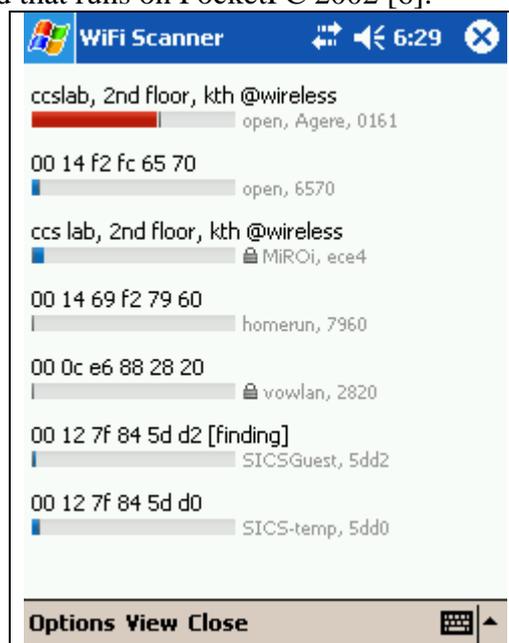
Figure 5. Herecast WiFi scanner. Associated access point is marked with red.

Figure 6. WiFiFoFum. Radar view conveys distance to access points.

Figure 5 and Figure 6 are screen shoots of Herecast [67] and WiFiFoFum scanners. These are just a few of the many tools that can be used to measure a WLAN's signal strength. Tool ranges from software which exists in most laptops to source code, from freely-available source code to very expensive products. [60, 61, 62,63,64, 65]

Before using these programs you should check if your network adapter is supported by them. Unfortunately promiscuous listening is not supported by Microsoft's Windows CE and the information needed to control the WLAN hardware has often not been available – so that developers could directly access this hardware. Thus it is sometimes necessary to add a WLAN interface which is supported by the software which you want to use.

1.3.3. Frame structure and frame types

IEEE 802.11 framing is more complicated than Ethernet framing because the wireless medium requires management features and corresponding frame types that are not found in wired networks.

There are 3 types of frames: Data frames, network management frames, and control frames. Data frames transport data from station to station. Control frames provide positive acknowledgment of received data and are used in conjunction with data frames to deliver data reliably. Management frames are used to join and leave wireless networks and move associations from access point to access point.

Here we will focus on the basic structures of 802.11 frames and the information they provide for network analysis.

1.3.3.1. Frame structure

Figure 7 shows a typical 802.11 frame. Not all the frames use all the address fields, and the value assigned to the address field may change depending upon the type of the frame.
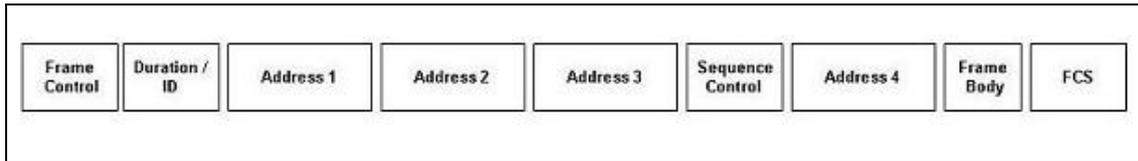
Figure 7. Typical 802.11 frame

Each frame starts with a frame control field and some of the components of this field are: Protocol version bit which indicates which version of 802.11 MAC is used. At present, only one version has been developed and it is assigned the protocol number 0. The type bit indicates the type of the frame. Wireless Equivalent Privacy bit (WEP) defines a set of encryption routines for protecting and authenticating data. When a frame is processed by WEP, this bit is set to 1.

Every frame may contain up to four address fields. Addresses are 48 bits long and if the first bit is a 0, the address represents a single station (unicast) and if this bit is 1, the address represent a group of physical stations and is called a multicast address. If all the bits are 1s, then the frame is a link local broadcast and is sent to all the stations.

Destination address is the MAC address of the final recipient. The source address is the MAC address of the source of a frame. If the receiver address is the MAC address of a wireless device then the receiver address is the destination address, but if the frame is destined to a node on an Ethernet connected to an access point, the receiver address is the wireless interface of the access point, and the destination address might be a router attached to the Ethernet.

The transmitter address identifies the wireless interface of the access point. The number of address fields used depends on the type of frame. Most data frames use three fields for source, destination, and BSSID.

The sequence control field is 16 bits and is used for both defragmentation and discarding duplicate frames.

The frame body, also called data field contains the higher-layer payload which is to be transported from station to station. IEEE 802.11 can transmit frames with a maximum payload of 2,304 bytes.

The frame check sequence (FCS) allows stations to check the integrity of the received frames. It is often referred to as a cyclic redundancy check because of the underlying mathematical operations used.

Data transfer is supported by the management frames and common subtypes are: Beacon frame, probe request frames, probe response frames, Association request frame, and Association response frame.

- *Probe request frame* A station sends a probe request frame when it needs to obtain information from another station. For example, a WLAN interface would send a probe request to determine which access points are within range. In addition some sniffing software such as NetStumbler sends probe requests to force the access points to sendthem the desired information.

- *Probe response frame* A station will respond with a probe response frame, containing capability information, supported data rates, etc., after it receives a probe request frame.

- *Association request frame* The network interface card (NIC) begins the association process by sending an association request to an access point. This frame carries information such as supported data rates and the SSID of the network it wishes to associate with. After receiving this request, the access point verifies that this information matches the parameters of one of the networks it supports (note that an AP may support several different networks).
- *Association response frame* If the AP accepts the association request from this NIC. Then information such as the association ID and data rate will be sent to the NIC.
- *Beacon frame* The access point periodically sends a beacon frame to announce its presence. This frame contains data such as timestamp, SSID, and other parameters regarding the access point to radio NICs that are within range. Radio NICs continually scan all 802.11 radio channels and listen to beacons as the basis for choosing which access point is best to associate with.
- After association, the station continues to scan for other beacons in case the signal from the currently associated access point become too weak.

Beacon frames announce the existence of a network and are an important part of many network maintenance tasks.

The destination address field in the header of the beacon frame contains all ones which is the broadcast MAC address. Thus all the devices within range of this access point will be forced to receive each beacon frame and process them.
The following information is provided by the frame body of the beacon frame:

- *Beacon interval* it is a 16-bit field set to the amount of time between beacon transmissions and is usually set to 100 milliseconds.
- *Capability information* stations must implement all the features in this field otherwise they cannot join the network. For example this information may require that all the stations must use WEP.
- *Timestamp* this is used by the station after receiving the beacon frame to update its local clock and synchronize with other stations that are associated with the same access point.
- Service Set Identifier (SSID) this identifies a specific WLAN.
- Supported rates: indicates the rates that are supported by the particular WLAN.

1.4. Indoor radio propagation

Radio propagation involves the transfer of energy between transmitter and receiver and this power is reduced on the way to the receiver because the signal looses power as it moves through the air and as it interacts with other material along the way. The way in which radio signals propagate must be taken into account when designing and operating radio networks, be they cellular telecommunication networks, WiFi, or other wireless technologies like Bluetooth.

The quality of the received signals is affected by the media which they travel through and the objects in their path. The radio waves can be reflected, scattered, and diffracted. They can even be absorbed and thus the distance over which they may travel will be significantly reduced.

Depending on the angle at which they arrive and the wave's frequency, radio signals are reflected when they encounter obstructions larger than their wavelength.

Radio waves are diffracted when they encounter edges, such as when they come into contact with the edge of a building. Scattering happens when the object dimensions are close to the signal wavelength and is frequently observed when the receiver or transmitter is located in a highly cluttered area.

Three different propagation phenomena can usually be distinguished: *multipath fading, shadowing* and *free space path loss*. The effect of these phenomena on radio propagation causes uncertainly and variability in the received signal strength indication (RSSI) values and makes it difficult to rely on a single measurement in order to calculate distance of propagation [56].

1.4.1. Free space loss

Received signal power is attenuated as a function of the distance from the transmitter and the most common form of this is called free space loss which means that the loss in signal strength that would result if affection of all absorbing, diffracting and reflecting influences were not taken into account on its propagation. As the name implies, free space loss assumes that the receiver and transmitter are both in free space.

1.4.2. Multipath fading

Radio signals reach the receiving antenna by two or more paths because of refraction and reflection from objects. A portion of the signal may go directly to the destination, and another part may bounce from a chair to the ceiling, and then to the destination. As a result, some of the signal will encounter delay and travel longer paths to the receiver. Also this will cause constructive or destructive interference and phase shifting of the signal. The receiving station will detect errors and will not send acknowledgement to the sender and the source will resend the packets.

This is the most problematic and difficult to overcome in WLAN networks. If the site survey software indicates that the WLAN is retransmitting packets often, then the network maybe experiencing multipath propagation.

1.4.3. Shadow fading

Persons walking across the line-of-sight between the device and an access point can degrade the link performance due to the high absorption loss of the person (or other material). The fast-changing shadowing situation may also trigger hand-over between access points. Many studies have been done about impact of human body shadowing on the performance of a high-rate wireless LAN system, see for example [7].

1.5. Positioning technologies

As this project is supposed to provide users with information about the WLAN coverage at the user's current location, an important part is estimating the user's position inside the building therefore in this section I will briefly introduce the most important technologies and research that have been done in this field.

1.5.1. What is the purpose of locating users?

Increasingly, IEEE 802.11 wireless network infrastructures exists in many places and one of the context parameters we can infer is the location of WLAN equipped

devices. Indirectly this gives us information about the possible location of the user of this device.

The main challenge in wireless networks has shifted from high data rates and capacity to services. Hence increasing interest in location technology for mobile devices. The desire to create location aware systems and services has lead to a strong motivation to develop techniques for estimating the position of wireless devices both indoors and outdoors.

A wide variety of applications have been developed that utilize location based technologies [8, 55]. Without having to disclose their location to others, users can utilize navigation oriented applications to display their location on a map, highlight local points of interests, or plot a route to a destination based on their current location. For example mappoint.com , streetatlausa.com, and dodgeball.com offer such services. [9, 10, 11]

The location information about the users can even be useful in optimizing cellular networks, for example a location-aided handover algorithm can use this information to select the most appropriate cell for handover, which leads to decreased signaling traffic and reduces the number of dropped calls. [12]

1.5.2. Cellular Network positioning

Location estimation can be done either by the mobile terminal itself or by the network, or some combination of the two.

In the first case the mobile terminal is responsible for both measuring the data and calculating the estimated location. In the other case the mobile terminal sends the data to the location server. The location server performs the calculations. In some cases the network can both collect the positioning data and perform the location computation.

In the simplest form we can use the identification of the serving cell or access point (in the case of WLAN) to find out the approximate location of the device, since we know that the, which is mobile terminal needs to be within range to be heard.

The accuracy of this method is proportional to the cell size in the network. For example in the GSM network the cell size varies from a couple of hundred meters to over 30 kilometers. This accuracy is not efficient for some tasks, particularly when the cells are large, but may be sufficient when the user simply wants to know if they are at home or at their office - which if they are sufficiently far aware from each other are unlikely to be in the same cell (see the application of Li Wei[13]).

One way to improve the accuracy is utilizing information about the signal strength because the mobile terminal is capable of measuring signal strength values. For example in GSM, the mobile terminal continuously measures the signal strength of up to 6 neighbor cells. The signal strength can be assumed to be proportional to the distance between the signal source and the mobile terminal and the location estimation can be solved geometrically. Signals from at least three cells are needed to solve the location of the mobile terminal.

The measured signal information can be compared to previously recorded values and then finding the best matching sample. This approach is called Database Correlation Method. [14] However, GSM has an even better method which is not so sensitive to variations in signal strength - this method is based on the **time advance parameter** - which is related to how far away the terminal is from the base station.

1.5.3. Satellite positioning

Global Positioning Systems (GPS) is the most widely used location systems for outdoor environments. However, it does not provide good accuracy inside buildings because GPS requires a signal path between the satellite and the receiver. Additionally, the GPS signal is too weak to penetrate many structures and the radio signals between the satellite and the receivers bounce around a lot disturbing the distance calculation. It also requires that every device be equipped with GPS hardware which increases its cost, weight, and power consumption. However, this is being added to many cellular phones and there exist zero chip GPS solutions which make use of the other radio receivers in the device and one chip solutions which add another receiver, but use the existing processor for doing all the necessary calculations.

For indoor positioning pseudolites (pseudo satellites) can be used.

1.5.4. Indoor positioning

1.5.4.1. WLAN based positioning

A WLAN based positioning system is an economical solution because most notebooks, PDAs, and other mobile devices are already equipped with WLAN interfaces and also because WLAN covers a larger area in comparison with other indoor positioning technologies (such as IR, ultrasound, etc.).

In WLAN a location-dependent parameter is the received signal strength indication (RSSI) as measured at the access point or at the mobile device. This value can be extracted from the 802.11 chipset through low level APIs.

There are two categories of such techniques, the empirical signal strength propagation model and the mathematical signal strength propagation model.

The idea behind both of these methods is that the signal strength received from an access point is a function of the receiver's location.

These methods usually work in two phases: *Off-line training phase* and *online location determination* phase.

During the off-line phase a so-called "radio map" is constructed before location positioning can begin. Some sample points in the area are selected (coordinates of these points are known) and at each point the signal strength value from the access points are measured and saved into a database. When a device requests its position, the signal information from all the nearby access points are collected and then sent to a machine which has access to the database for the comparison.

Empirical and propagation models differ in the way they construct the radio map. The empirical model based systems construct the map by collecting signal strength measurements and recording the observations. The propagation model based systems use a mathematical model of the degradation of signal strength over distance to compute signal strength values and record the results in the radio map.

1.5.4.1.1. Empirical Model

In this method, when a network administration tries to find a wireless device, each access point replies with the signal strength of the devices it hears then the location

tracking system takes this information and matches it against the database of locations. When it finds a match, it is reported.

There are two disadvantages to the systems built using this method. The system always requires a considerable amount of effort to construct the radio map and populate the database. Both have to be done for each new environment. The other disadvantage is that the system will lose some accuracy when the current environment is changed, for example when desks are moved, base stations are moved, doors are opened and closed, etc.

Some of the positioning systems based on this method are: RADAR [15] and AMULET [16].

1.5.4.1.2. Propagation Model

The idea behind the propagation model is that signal strength decreases with distance from the access point. How much it decreases depends on the properties of environment.

For example when the IT administrator initiates a command to find a wireless device all the access points on the network, for example when desks are moved, base stations are moved, doors are opened and closed, etc.

Access points that do not hear the device do not respond. The IT administrator waits to ensure that all the access points have had the opportunity to respond. The more access points that respond the greater the accuracy will be.

The location tool then draws circles around the access points that that hears the device. The smaller the circle, the stronger signal. Algorithms are used to determine the most likely location of the device within the intersections of these circles.

This method does not take into account characteristics like reflection, attenuation and multi path effects.

Positioning systems like RADAR and Horus [17] use this method.

1.5.4.2. Closest Access Point Method

The closest access point method finds devices within range of a single access point. It is the simplest and least accurate way to locate a device (and the associated user); however, it is the simplest method.

We have to access the low level information of the device such as the MAC address of the access point we are connected to or the name of the access point. Then we could use a database containing information about the location of all these access points to obtain the location of the user.

Using this method an IT manager can send a request to the system asking for the location of "Ann" and the system returns the access point that Ann is currently associated to. [18]

1.5.4.3. Bluetooth

Bluetooth is a technique for short-range wireless connectivity. It is widely used as a cable replacement to connect mobile phones, computers, and other digital devices via a 2.4 GHz short-range radio link. As with WLAN, the measured signal strength from several Bluetooth devices can be utilized to estimate the location, however, as noted

earlier the signal strength measurement is very poor - but since the range is short, you can know roughly where the user is by using the closest access point method (assuming that you have Bluetooth access points). [19]

1.5.4.4. Location through SNMP

SNMP (Simple Network Management Protocol) [20] is an application-layer protocol that facilitates the exchange of management information between network devices.

Basically SNMP provides a set of rules that allows a computer to get statistics from another computer across a network. This enables network administrators to manage the network, measure performance, and find and solve network problems.

A location system can be implemented by sending periodical queries to the managed devices. These queries will collect information such as the list of hosts associated with access point. The information can be requested periodically from the access points and saved into a database so that answers can be given quickly. Also this method requires no additional software in the clients and only SNMP software in the server.

The disadvantage is that although SNMP is a standardized protocol, not every access point supports it, the access point information is usually stored in enterprise-private variables, this making it difficult to get the desired information and to interpret it.

1.5.5. Server-Oriented vs. Client-Oriented

The user's location could be estimated either in a server or in the user's device. If the positioning task is done in the server, then users could access a webpage and see information related to their location. This approach requires a method of mapping a client's IP address to a physical location and this requires collaboration with the administrators of the wireless network. The IP to MAC mapping requires SNMP access to the gateways and the MAC address to access point mapping requires access to the Radius server's log file. Once the client's current access point is known, then the information related to that location can be delivered.

Note that knowing only the MAC address of the device cannot help determine its location in the case of mobile users.

Alternatively, we could let the device determine its own position. In the simplest case the software running on the client determines the near-by access points (i.e. the access points it can hear), then the client could look up the location of these access points in a database that gives every access point's actual location in world coordinates. This method has a privacy advantage because the client does not need to reveal its location.

In my project I will utilize a client based model because this best supports my goal of giving individual users services based on their location.

1.5.6. Prior work

1.5.6.1. The RADAR location system

The RADAR location system is based on utilizing IEEE 802.11 wireless networks. In this architecture the position estimation occurs in two phases, offline and on-line. It is an example of a client-server-based location mechanism. A program installed in every client scans the network and measures the received signal strength (RSSI) of every AP and then sends a vector with all of these RSSIs to the location server. The server receives this vector and matches it with a radio propagation map stored in a database during the earlier offline phase. In this figure you see the structure of such a system.

The main problem with map-based techniques is the calibration in the offline phase [2]. The accuracy of such systems depends on this procedure that consists of physically moving a wireless device to each location in the map and capturing RSSI values from APs.

The Ekahau positioning engine [21] is one of the commercial positioning systems based on IEEE 802.11 wireless LAN and determines the location of the devices based on a detailed computation of radio propagation in an environment.

Additionally these map based systems suffer from the problem people and other things moving about in the environment, since unfortunately this changes the multipath propagation and hence changes the signal strength which is received. So the results are really only correct when there are no users!

1.5.6.2. LOCADIO

LOCADIO [22] uses WiFi signal strengths from existing access points measured by the client to infer the user's location and to determine whether or not the user is in motion.

For motion, it measures the variance of the signal strength of the strongest access point as input to a simple two-state hidden Markov model (HMM)1 for smoothing transitions between the inferred states of "still" and "moving." For location, it exploits the fact that WiFi signal strengths vary with location, and it uses another HMM on a graph of location nodes whose transition probabilities are a function of the building's floor plan.

1.5.6.3. The Horus WLAN Location Determination System

Horus is a based on an empirical model and its aim is to satisfy two goals: high accuracy and low computational requirements. [17]

In the offline phase the radio map will be constructed and the location of the user will be estimated based on the received signal strength from each access point and the radio map prepared in the offline phase.

The Clustering module is used to group radio map locations based on the access points covering them. Clustering is used to reduce the computational requirements of the system. The Discrete Space Estimator module returns the radio map location that

---

[1] For more information about the Hidden Markov Model see:
http://en.wikipedia.org/wiki/Hidden_Markov_model

has the maximum probability given the received signal strength vector from different access points. The Correlation Modeling and Handling modules use an autoregressive model to capture the correlation between consecutive samples from the same access point.

This model is used to obtain a better discrete location estimate using the average of n correlated samples.

The Continuous Space Estimator takes as an input the discrete estimated user location, one of the radio map locations, and returns a more accurate estimate of the user location in the continuous space. The Small-Scale Compensator module handles the small-scale variation characteristics of the wireless channel.

1.5.6.4. Place Lab

Place Lab is an open source toolkit for device positioning for client-side location based computing and works both indoors and outdoors [23]. It was developed by researchers at the University of Washington, University of California at Berkeley, and Intel in October 2004.

The Place Lab architecture consists of three key elements: Radio beacons in the environment, databases that hold information about beacons' locations, and the Place Lab clients that use this data to estimate their current location. A short description for each of these elements is given below.

Beacons could be WLAN access points, fixed Bluetooth devices, and GSM cell towers. Each of these is assigned a unique ID. The coverage and accuracy of Place Lab depends upon the number and type of beacons within range of the client device.

The beacon database(s) come from institutions that own a large number of wireless networking beacons. At this time, Place Lab clients have access to location information for approximately 2.2 million radio beacons, primarily IEEE 802.11 access points. These mostly come from wigle.net, but more accurate databases exist for University of California at San Diego and the University of Washington, as well as some GSM tower locations imported from the FCC's database. To allow others to experiment with beacon placement algorithms, they also maintain a database that currently has location estimates for around 40,000 GSM, IEEE 802.11, and Bluetooth beacons. The Place Lab clients use live radio observations and cached beacon locations to estimate their location. Clients fall into one of three classes: spotters, mappers, and trackers.

Place Lab is written entirely in J2ME and currently runs on the Windows XP, Linux, Pocket PC 2003, and Symbian.

Place Lab's clients instantiate one spotter per radio protocol supported by the device. For example a laptop might have a Bluetooth and an IEEE 802.11 spotter, while a cell phone might run a Bluetooth and GSM spotter. The spotter's task is to monitor the radio interface and share the ID's of the observed radio beacons with the other system components.

The mapper provides the location of known beacons and may obtain this data directly from a mapping database or from a previously cached portion of a database. In the last step the tracker uses this data to estimate the user's position

1.5.6.5. Community-Built Database

At Worcester Polytechnic Institute they have created an access point location database that can support an entire campus, city, or even the world. The first time an access point is discovered by a user, that user can enter information describing the location of this access point. This information may then be uploaded to the global database and made available to anyone else that happens to encounter that access point.

The database knows the physical location of each access point: its country, city, street, number, building, floor, and room.

So every time the user is associated with a specific access point from the database the location of the access point and the user merge with previous paragraph be obtained.

For efficiency a portion of database can be downloaded and cached by the client when it enters a new area, so that the user will be able to move around in an area without having to download more information for every access point that is encountered.

Explain why this model of caching (which is also exploited with tiled models in games and even physical maps is so powerful.

1.6. Related work

Currently there are not any indoor visualization projects running. However there are many not detailed WiFi maps for outdoors. There are some wireless mapping projects in which a symbolic markup on the map indicates the location of access points. These maps are maintained by wardrivers and are only useful for visualizing the number and general layout of the network and access points. Exact coverage cannot be determined from the recorded data [57, 58]. Thes2 project [59, 66] visualizes outdoor WLAN in more details and shows the extension of individual access points on the map.

In this chapter I will discuss some visualization projects for outdoors and some recent indoor positioning projects as there is not any detailed approach for indoor models.

1.6.1. WiFi map, Silver Lake Ohio

The position of access points have been captured by GPS enabled NetStumbler and then integrated to the Google map using Ajax, XML and Google API [24].

1.6.2. UIC Wireless Access points, University of Illinois at Chicago

As you see in the Figure 8 , you click on a specific building on the map and you will be prompt by information about the coverage in that building, such as which floors have coverage and which floors do not have.

For example in the 5th and 7th floors of the Medical Science building some access points are installed.

1.6.3. University of California, Campus Modeling and Visualization

The goal of this project at the University of California [25] is to build an accurate environment model to be used in outdoor Augmented Reality (IR)[2], providing an interface to support navigation through the campus, and construct a framework for research in 3D visualization.

Each building model is exported into a VRML file [3] and then each 3D model is aligned with the campus CAD files. In the last step elevation information and texture is added to the model.

1.6.4. iSPOTS

The aim of this project is to create a better understanding of the daily working patterns of the academic community by monitoring and collecting data of WiFi usage on the Massachusetts Institute of Technology campus [26].

The WiFi coverage of the campus will be mapped onto a three dimensional plan of campus and compared with the plan of antenna locations. Then, the log files of antenna traffic will be employed to monitor the use intensities of all WiFi ispots in real time. This data will be used to construct an electronic color map, which shows the WiFi coverage and intensity use on a web page.

This study will analyze which areas of the campus are most accepted among students for study and work and which are the criteria for choosing among locations. Results will also be used for developing a better understanding of building's use pattern, designing new working environments, and determining the priorities for providing future WiFi infrastructure.

---

[2] IR means combination of real world and computer generated data [27].
[3] Three-dimensional world, scene, or object based on the Virtual Reality Modeling Language (VRML); may be a standalone 3D file or may be integrated into a Web page; most VRML files let the user navigate the 3D world using the keyboard and/or mouse [28].
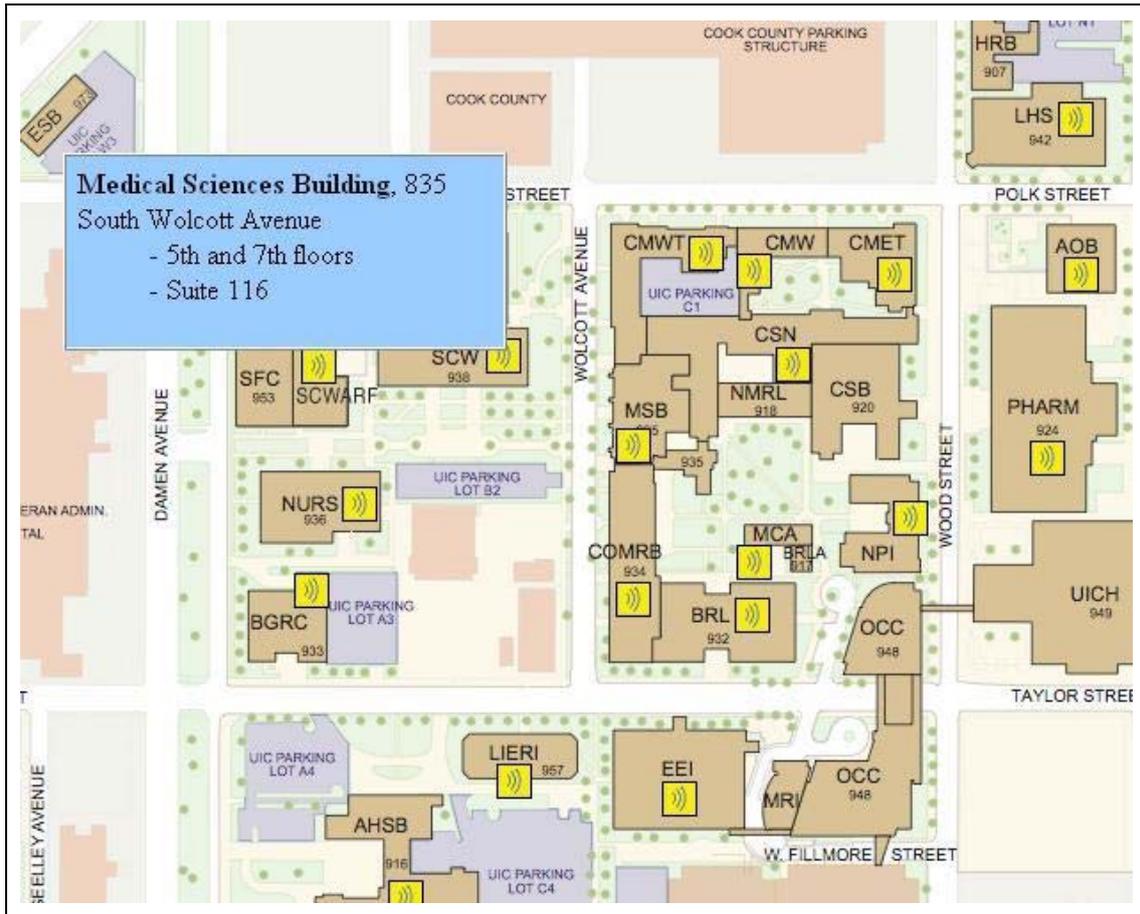
Figure 8. UIC Wireless Access points

1.6.5. Texas A&M Campus Navigator

Web-Based Route Planner for Texas A&M (TAMU) Campus Area

This application enables users to quickly find directions to all TAMU buildings, departments, and major services [29].

The system has five main modules. The client side interface is implemented using the Google Maps API [30] written in Java Script. It communicates to the web server using AJAX [31]. An Apache web server is the main coordinator of the server side modules. The path planner is written in C++ and computes paths and other displayable information using a modified version of Dijkstra's shortest path algorithm on a hand made layered roadmap. All roadmap and landmark information is stored in the database. The visualizer is Google Maps [32], which provides satellite, map level, and a hybrid view of the campus the client to display this data overlaid with paths and other queried information. See Figure 9.
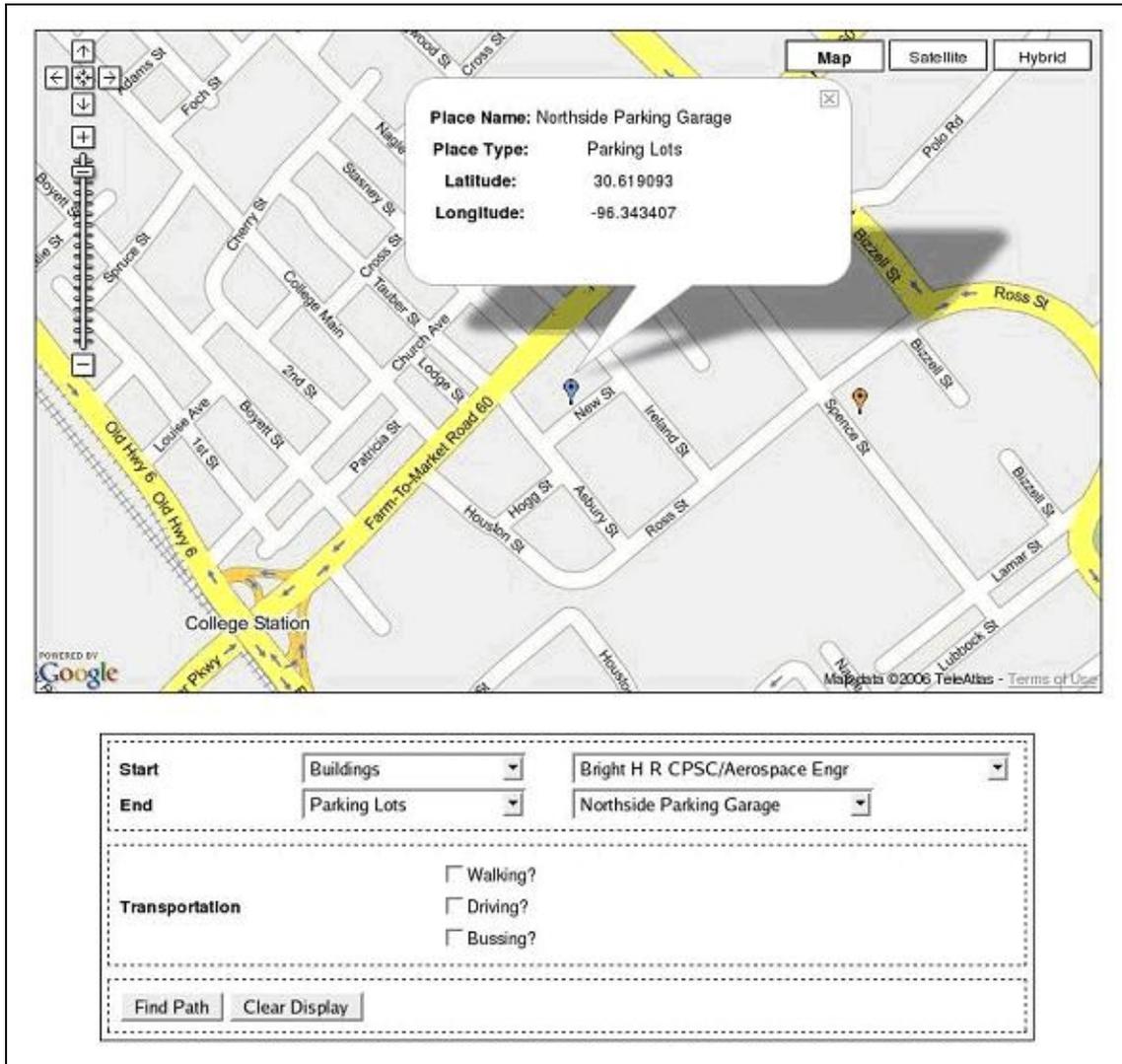
Figure 9. Screen shot of Texas A&M campus navigator

1.6.6. Campus Navigator, University of Victoria

This application [33] which works both on laptop and Pocket PC enables new students to find out where they are located and how to get to a certain destination on the campus.

The WRAPI library [34] has been used for getting information about the IEEE 802.11 network.

The main part of the graphical user interface is the campus map which contains a green oval that indicates the current location of the user. This location is determined when the user starts the program. The program runs a scanner module to obtain the MAC address of the access point with the strongest signal strength. This result is then used to search a database to find which building the user is in.

Once the user has selected the building he or she wishes to go, there is drop down menu with a yellow arrow appearing over that location on the map. See figure 14 for more details.

1.6.7. ActiveCampus, University of California, San Diego (USCD)

The UCSD ActiveCampus project is an exploration of wireless location-aware computing in the university [35]. ActiveClass supports classroom activities such as anonymous asking of questions, polling, and student feedback. ActiveCampus Explorer supports several location aware applications, including location-aware instant messaging and maps of the user's location etc. This figure shows a screenshot of this application.

1.6.8. Locus, Worcester Polytechnic Institute (WPI)

Locus is a tool developed for indoor location sensing system using location finger printing techniques. Additionally is compares the performance of various proximity-matching location fingerprinting algorithms (Several proximity-matching algorithms can be used to compare the stored signal strength values to the observed real time values to infer the user's location.) and presenting a location sensing module for building context-aware applications on WLANs with display in Scalable Vector Graphics (SVG) [36].

Locus has been designed in Java in order to meet the platform independent goal. It uses Wireless Research API (WRAPI), a module for retrieval of signal strengths.

WRAPI is developed in C++ for the Microsoft Windows XP operating system. To separate this system dependent component from the rest of the system, a Java Native Interface (JNI) class was created to handle the communication.

The Locus graphical interface uses the Batik API [37, 38, 39] for manipulating SVG files and content. Figure 16 displays the look of Locus.

1.6.9. Aura, Carnegie Mellon University

The fact that resources in a particular environment may change dynamically and frequently causes mobile users problems managing their computing resources in each new environment. The project Aura [40] at Carnegie Mellon University is a new solution to this problem.

The main challenge in such solutions are maximizing the use of available resources (that is effectively exploiting the computing and communication resources in the environment) and to minimize user distractions.

The idea behind Aura is that it acts as a proxy for the mobile user it represents: when a user enters a new environment, his or her Aura marshals the appropriate resources to support use's task. Example of user's tasks can be writing a paper, preparing a presentation, etc.

To provide applications with environmental information a Contextual Information Service has been developed to provide properties of both physical entities and available resources such as: the location of people, the location and properties of printers, the amount of network bandwidth available, etc.

Requests such as: When bandwidth will be best, within the next hour, to flush the distributed file system's cache; or where the closest color printer with an empty print queue is are supported.

Currently two version of Aura in Java and C are implemented.

2. Introduction to Microsoft Windows mobile development

2.1. Mobile Operating Systems

Some of the common mobile operating systems are:

Windows Mobile	*designed for devices like Dell, HP and Motorola, with different wireless operators. These devices are available on GSM or CDMA networks. [41]*
Symbian OS	*a standard operating system for smart phones and is designed for 3,5G and 3G mobile phones. [42]*
Palm OS	*launched in 1996 for the first time and provides the mobile devices with capability to access the Internet via a wireless connection.*
Linux and Unix	*Linux and other UNIX systems have been used to provide wireless connectivity to mobile devices since the early 1990s. [44]*

2.1.1. Windows Mobile

In 1996 Microsoft launched the first version of Windows mobile (Windows CE 1.0) to compete with Palm. Windows CE is a compact operating system for mobile devices based on the Microsoft Win32 API. Devices such as the Pocket PCs and Smartphones run this operating system.

Some of the platform capabilities are:

Multi-threaded tasking	*other competitors like Palm OS provides a simpler, power-conserving operating system but CE designers opt for more powerful solutions. Batteries can on Palm devices last up to six weeks but Windows CE devices can drain batteries within a single day.*
Power management	*the Windows CE API does support a set of power monitoring functions in order to allow applications to determine the remaining life of the battery. Well-written CE applications will monitor power levels and will alert users and gracefully exit as levels become critically low.*
Network communication	*Windows CE devices support standard Ethernet connections as well as WLAN.*

Figure 10 shows Windows mobile 5.0 for Pocket PC.

Figure 10. Windows mobile 4.2

Windows mobile 2003 was released on June 23, 2003 and uses Windows CE 4.2.0. Windows mobile 2003 Second Edition was released on March 24, 2004 and uses Windows CE 4.2.1. It includes a number of improvements over its predecessor, such as the option to switch from landscape to portrait screen layout.

Windows mobile 5.0 was released on May 9, 2005 and it uses Windows CE 5.0. It provides some new features such as a new version of Office (called Office Mobile which includes PowerPoint Mobile, Excel Mobile, and Word Mobile), Windows Media Player 10 Mobile, Global Positioning System management interface to all navigation programs, and an error reporting facility similar to that present in desktop Microsoft Windows systems.

2.2 Development tools

The Microsoft .NET Framework is a software framework that can be added to the Microsoft Windows operating system. It is an environment for building, deploying, and running Web Services and other applications. It consists of three main parts: the Common Language Runtime, the Framework classes, and ASP.NET.

The .NET Compact Framework is a subset of .NET Framework and is designed for developing applications for Pocket PC, smart phones, and smart devices. It uses some of the same class libraries as the .NET Framework and also a few libraries designed specifically for mobile devices.

Microsoft's Visual studio 2005 is a single integrated development environment (IDE) for native, managed, and server-side applications for Windows Mobile 2003, Windows Mobile 2003 Second Edition, and Windows Mobile 5.0 platforms. Earlier devices are supported by using Microsoft's embedded Visual C++ or visual studio 2003 for native and managed code development respectively.

Windows Mobile platforms are supported can be targeted natively with Visual C++ and .NET Compact Framework and for managed code by using C# or Visual Basic .NET.

2.3 Microsoft SQL Sever CE 2

SQL Server CE 2.0 [48] is Microsoft's compact database server that has a small enough footprint (~1.5MB) to fit and run in a Pocket PC device. It is included in Visual Studio 2005, so as a developer we don't have to install it.

For end users, SQL Mobile can be installed on the Pocket PC 2003, Windows Mobile 5.0, Smartphone 5.0, and Windows CE 5.0 platforms. This server was used in this project to insert the measured data in that for further analysis.

2.4 SQL Server CE Query Analyzer

SQL Server CE Query analyzer is a graphical tool included in Microsoft SQL Server CE that can be used to create databases, create and modify tables in a database, query objects by using the SELECT statement, insert and delete rows in a table, etc.

I used this tool to check that data has been correctly inserted to the database while I was scanning the area and later the scanned data were transferred to the desktop computer for further analysis. Figure 11 is as example of inserted data into a database.

Figure 11. Collected data viewed by Query analyzer.

2.5 OpenNETCF

OpenNETCF.org is a project run by OpenNETCF Consulting. The primary product of this project is the Smart device Framework which is a set of classes that complement the .NET Compact Framework and provides access to many functions which are very useful for developers - hence reducing development time.

In Windows, interaction with wireless devices is performed by using the Network Device Interface Specification (NDIS) User-mode Input/Output (NDISUIO) to query an NDIS miniport driver through DeviceIoControl calls. These are difficult to

implement in C# because they require using P/Invoke call to access unmanaged dlls. However OpenNETCF makes it possible to query the wireless device driver parameters through a managed OO wrapper around NDISUIO.

The main NDISUIO query of interest in this project is the BSSID List Scan, which returns a list of all neighboring access points, the RSSI values, quality of received signals, MAC address of the access points, and the SSID of the associated access point.

2.6 HP iPAQ 5550 Pocket PC

The HP iPAQ 5550 runs Pocket PC 2003 OS and has an Intel XScale PXA255 processor running at 400 MHz. It has a built-in WiFi IEEE 802.11b WLAN interface. The nub on the top of the device is the WiFi antenna. See Figure 12 .

Figure 12. HP iPAQ 5550

Further detailed specifications of this PDA are shown in Table 2:

Table 2. HP iPAQ 5550 characteristics.

Display	Transflective TFT color LCD, 65,536 colors, screen size diag: 3.8", resolution: 240 x 320.
Battery	Lithium Ion Polymer rechargeable. Battery is user replaceable. 1250mA.
Performance	Intel XScale PXA 255 400 MHz processor.
Size	5.23" (not including antenna nub) x 3.3" x .63". Weight: 206.5 g.
Operating system	Windows CE 4.2
Networking	Integrated WiFi 802.11b (also supporting LEAP) and Bluetooth.

2.7 Visualization technologies

2.7.1 Google Earth

Google Earth is a virtual globe program that maps the earth by images obtained from satellite imagery and geographical information system (GIS) over a 3-dimentional globe [49]. It was developed by Keyhole, Inc., a company acquired by Google in 2004. The product was renamed Google Earth in 2005 and is currently available for use on personal computers running Microsoft Windows 2000 or XP, Mac OS X 10.3.9 and above, or Linux (released on June 12, 2006). Google Earth is a tool for viewing, creating, and sharing location-specific information which can be

explored via an interactive and visually intuitive interface. In addition it allows users to search for addresses (for the USA, Canada, and Europe only), enter coordinates, or simply use the mouse to browse to a location. Images were taken by satellites in the last three years and are updated on a rolling basis. Figure 13 shows a screenshot of Google Earth that displays the location of the Wireless@KTH research center in Kista, Stockholm- where this thesis project took place.
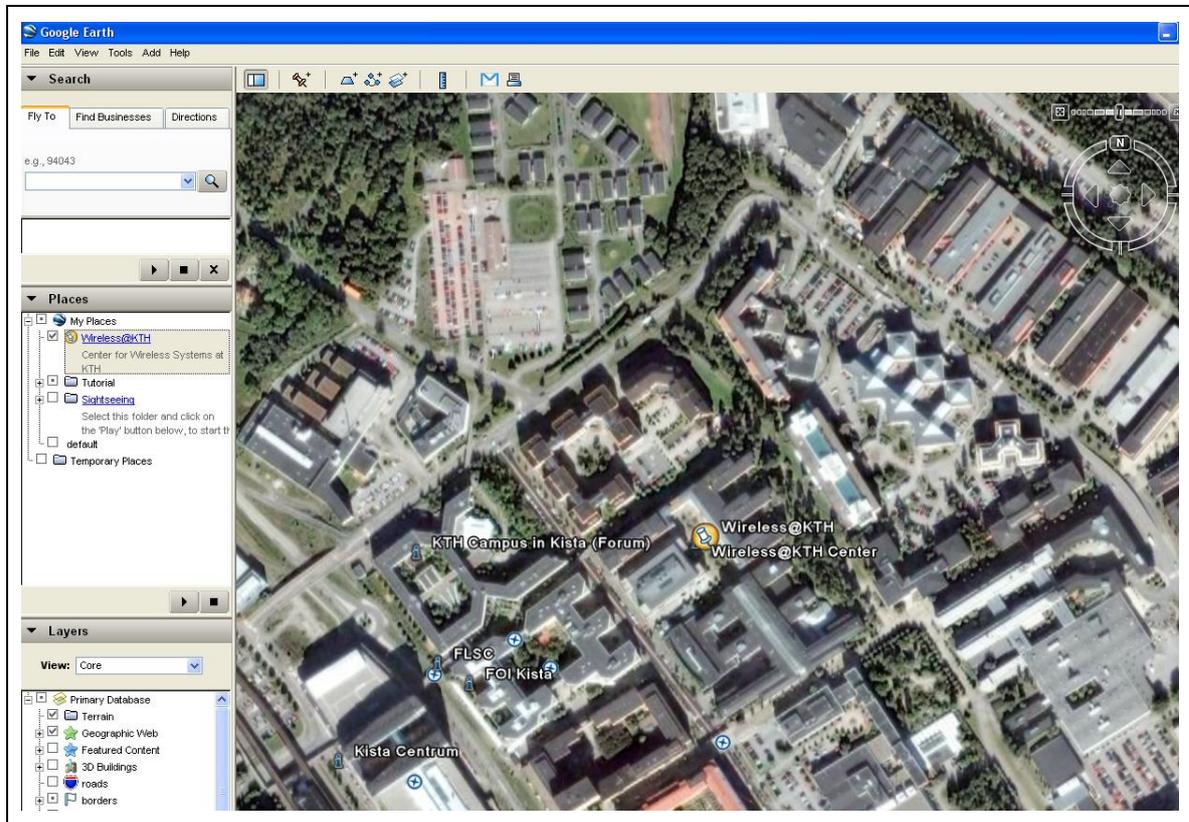

Figure 13. Google Earth

There are currently three versions of Google Earth:

- The free version for home/personal use
- The enhanced version for home/personal use which costs US$20 /year
- The professional version for commercial use which costs US$400 /year

The last two versions support GPS data import, real-time GPS tracking, better image quality, higher resolution printing, customer supports via email, and reading address point from Comma-Separated Values (CSV) file.

The professional version is a business-oriented upgrade to Google Earth that additional features, including add-on software such as movie making, GIS data importer, and advanced printing modules.

On January 8, 2007 the latest version of Google Earth (Google Earth 4) was released which has the additional feature of higher quality terrain data for many mountainous regions, as well as support for textured 3D buildings.

Millions of Google Earth users add and share geographic information every day using KML (the Google Earth file format). In Google Earth 4 it is even possible to

view travel photos; hiking trails (based on the GPS data) and even time-stamp the data to create animations of how the world changes over the time.

System requirements for Google Earth on the PC are:

Operating System	Windows 2000, Windows XP
CPU	Pentium 3, 500 MHz - System Memory (RAM): 128MB RAM
Hard Dis	400MB free space
Network Speed	128 Kbits/sec
Graphics Card	3D-capable with 16MB of VRAM
Screen	*1024x768, "16-bit High Color" screen*

2.7.2 Google SketchUp 6

SketchUp [50] is a program that enables professional architects, civil engineers, filmmakers, game developers to build, view and modify 3D models quickly and easily. It was designed to be more intuitive, flexible, and easy-to-use than other 3D modeling programs. It was first released in August 2000 as a general purpose 3D content creation tool. Then it quickly found a market in architecture and building design industries. The key to its success has been the short learning period in comparison with other commercially 3D tools.

The latest version (SketchUp 6, released in 2007) allows users to place their models into the Google Earth using real-world coordinates and share them with the world. Some of its key features include: a smart drawing cursor (inference) system that allows users to draw 3D objects using a 2D screen and mouse; models can be individually colored with an assortment of solid colors, textures, and materials; and the ability to trace a photo to create a 3D model.

Figure 14 is a router and is an example of what can be done by SketchUp.
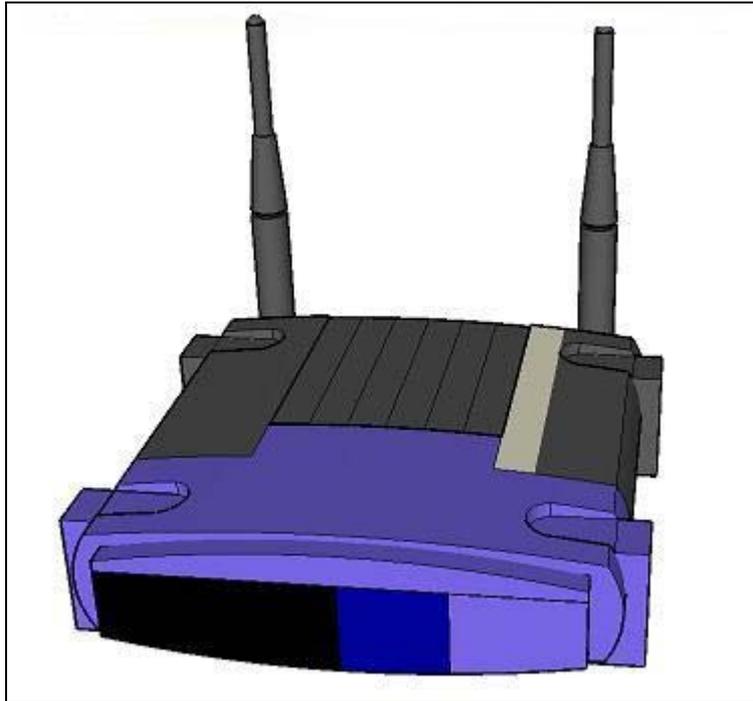
Figure 14. A wireless router done in SketchUp.

2.7.3 Scalable Vector Graphics (SVG)

SVG is a graphics standard developed by the World Wide Web Consortium (W3C) for describing two-dimensional graphics in XML. SVG inherits all XML capabilities (such as interoperability), and provides tools to create geometric objects, and easy manipulation and transformation. These attributes make SVG a suitable tool for map visualization interface.

Unfortunately, current web browsers do not support SVG natively, but a plug-in or some SVG viewing software can be used. As an example, Adobe's SVG viewer is a freely available plug-in [45]. There are even some SVG viewer available implementations for mobile phones and PDAs. [46]

The wireless coverage application is an internet application project at the IT department of Rochester Institute of Technology (RIT). It uses SVG to show the wireless networking coverage zones for all the buildings at RIT. This tool can be used to help understand areas without WLAN coverage and to facilitate future planning of access point placement. [47]

2.7.4 Ruby

Ruby [68] is an open source object oriented programming language which was created by Yukihiro Matsumoto in 1995. He has blended Perl, Smalltalk, Eiffel, and Lisp to form a new language that balances functional programming with imperative programming.

2.8 summaries of techniques

A program written in Visual Studio 2005 environment using C# and the OpenNETCF library will be used to scan the network. This program will be runned in a HP iPAQ 5550 and the scanned data will be inserted into the SQL Server 2.0 installed inside this device.

27

SQL Query Analyzer will be use to analyze the data. Further SVG maps, Ruby, Google SketchUP, and Google Earth will be used to visualize the wireless coverage

3. WLAN signal propagation analysis

3.1 WLAN scanner application

In order to scan the WLAN an application was implemented using the OpenNetCF.net namespace. A list of access point MAC addresses (one for each access point), the associated received signal strength indication (RSSI), and supported rates for every access point will be retrieved by this application.

RSSI is normally measured in decibels referred to one milliwatt, or dBm. The usual range is approximately -100 to -20 dBm. The APIs we use report RSSI as an integer - the translation between this integer and an actually dBm measurement is vendor dependent.

RSSI generally decreases with distance from the access point, but it is also affected by attenuation and reflection, making the relationship between location and RSSI complex.

The supported rates entry is an array in kilobits per second indicating that the radio is capable of running at the indicated rates.

3.1.1 Adapter class

The Adapter class was used to retrieve configuration information about network adapters, be they PCMCIA cards, USB network cards, built-in cards, etc. Information retrieved might be the IP address of the adapter, currently attached SSID (i.e., the name of the current access point), the hardware address associated with the adapter, a list of SSID values which the adapter can currently hear, etc.

A list of all adapters that a device has is listed in a combo box by following C# code as shown in the Figure 15.

```
// get the available adapters
m_adapters = Networking.GetAdapters();

// clear the combo
cboAdapters.Items.Clear();

// add the adapters
foreach (Adapter adapter in m_adapters)
{
    cboAdapters.Items.Add(adapter);
}
```

Figure 15. Retrieving adapters.

3.1.2 Access point and AccesspointCollection classes

Access points and their properties such as MAC address, signal strength, and the SSID name can also be retrieved by using these classes. For example to retrieve the list of access point and for each of them the indicated information, we simply do as shown in Figure 16.

Figure 16. Retrieving access point properties.

The GetRates method returns an array of supported signaling rates in kilobits per second for the adapter. See a screenshot of this program in Figure 17.
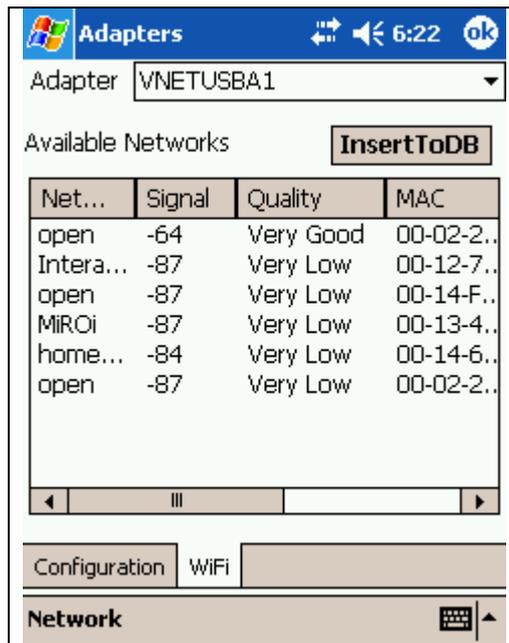
Figure 17. WLAN scanner application.

3.1.3 Wifidb database

Table 3. Wifidb table. shows the ScannedAPs table from the wifidb SQL CE database which I have created for collecting information about access points and their properties.

Table 3. Wifidb table.

Network	Signal	Quality	MAC	SupportedRates	Time	Associated	Scannr

Note that the SQL server used in this application is SQL CE 2.0 (See section 2.3) is running on the mobile device and the measured data can be viewed by using Query Analyzer. (See section 2.4)

Network	the SSID.
Signal	the signal strength in decibels along with a text description like Excellent, Good, Fair or Poor.
The associated access point	the access point that is currently running the network traffic. It is possible to have no associated access point.
Scannr	the number of scan.
Time	indicates the time which scan was performed.
Supported rates	the rates supported by the network.

The loop in Figure 19 inserts the measured data into the ScannedAPs table of the wifidb after that the user has clicked the "InsertToDB" button. See Figure 18. Inserting rows into the database..
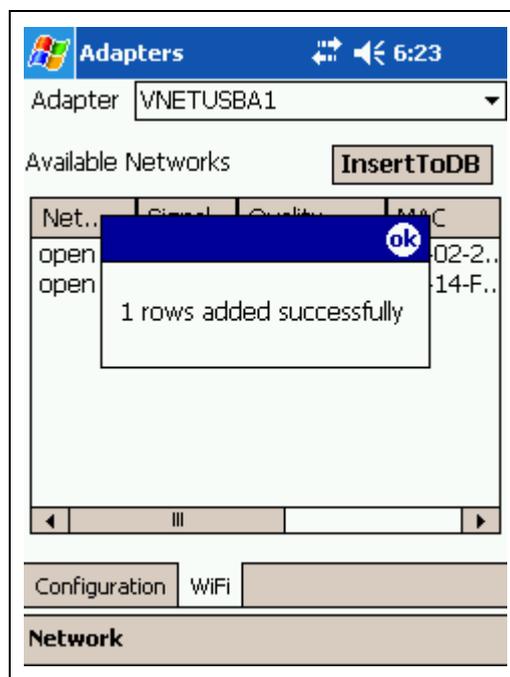
Figure 18. Inserting rows into the database.

```
cmd.Parameters.Add(new SqlCeParameter("network", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("signal", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("quality", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("MAC", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("associated", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("rates", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("time", SqlDbType.NText, 50));
cmd.Parameters.Add(new SqlCeParameter("scannr", SqlDbType.NText, 50));


int t = 0;
for (t = 0; t < lvwNearbyAPs.Items.Count; t++)
{
    cmd.Parameters["network"].Value = lvwNearbyAPs.Items[t].SubItems[0].Text;
    cmd.Parameters["signal"].Value = lvwNearbyAPs.Items[t].SubItems[1].Text;
    cmd.Parameters["quality"].Value = lvwNearbyAPs.Items[t].SubItems[2].Text;

    cmd.Parameters["MAC"].Value = lvwNearbyAPs.Items[t].SubItems[3].Text;
    cmd.Parameters["associated"].Value = lvwNearbyAPs.Items[t].SubItems[4].Text;
    cmd.Parameters["rates"].Value = lvwNearbyAPs.Items[t].SubItems[5].Text;
    cmd.Parameters["time"].Value = DateTime.Now;
    cmd.Parameters["scannr"].Value = scannr;

    cmd.ExecuteNonQuery();
```

Figure 19. Inserting rows into the database.

Figure 20 shows an example of a survey that has been done in the second floor of 2IT building using this application. The collected values have been retrieved through Visual Studio 2005.
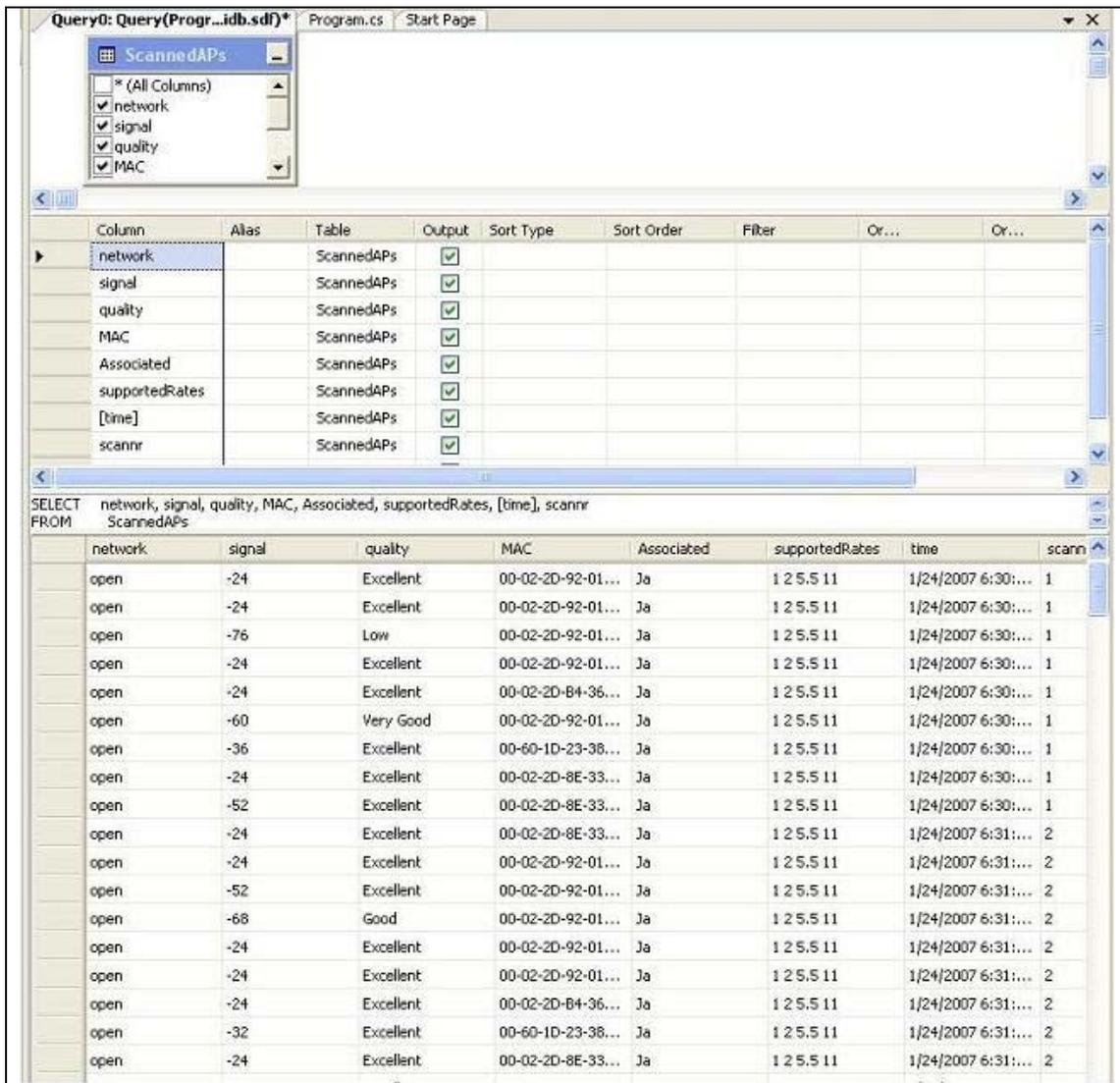
Figure 20. Sample of collected data.

3.2 Experimental environment

In this part, the setting up of a test bed to measure and analyze the characteristics of The WLAN wave signal will be discussed.

The test bed for WLAN signal analysis is shown in Figure 21 and is the 4rth floor of the Forum building in Kista. This is a university building containing hallways, lecture halls, computer rooms, and a cafeteria.

Three Cisco Aironet1200 and Cisco AP1000 access points are installed on the roof of this floor as you can see in the figure 22. (Circles indicate the position of APs)

To study the WLAN propagation, we use HP iPAQ 5550 Pocket PC running our own software to detect the WLAN APs nearby. We used our application to scan access points and collect the RSSI associated with each access point.
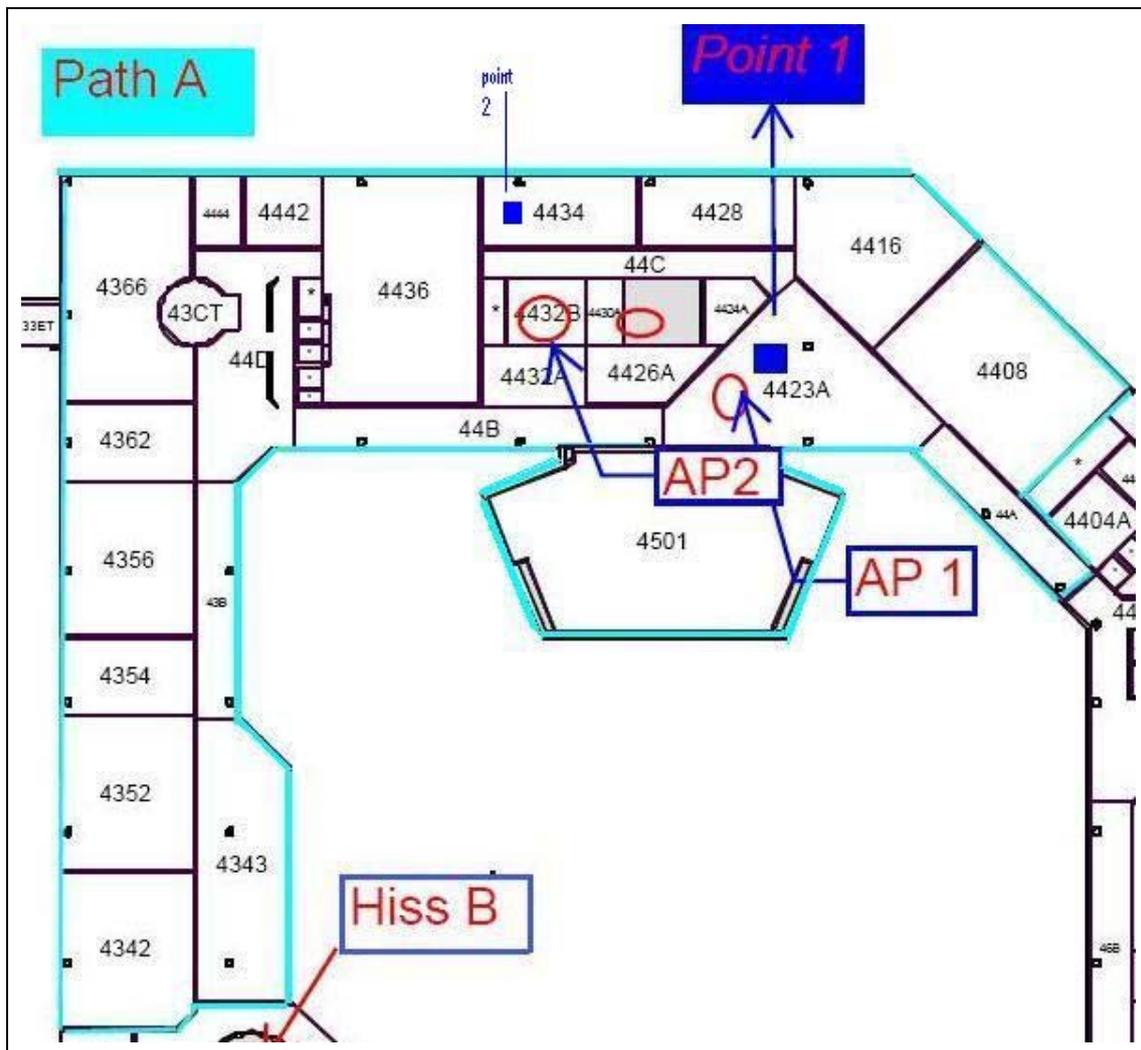
Figure 21. The experimental area.

3.3 Analyzing the characteristics of the WLAN signals

3.3.1 Signal distribution of one access point

Figure 22 shows the resulting coverage map from access point AP2 in the testing area.
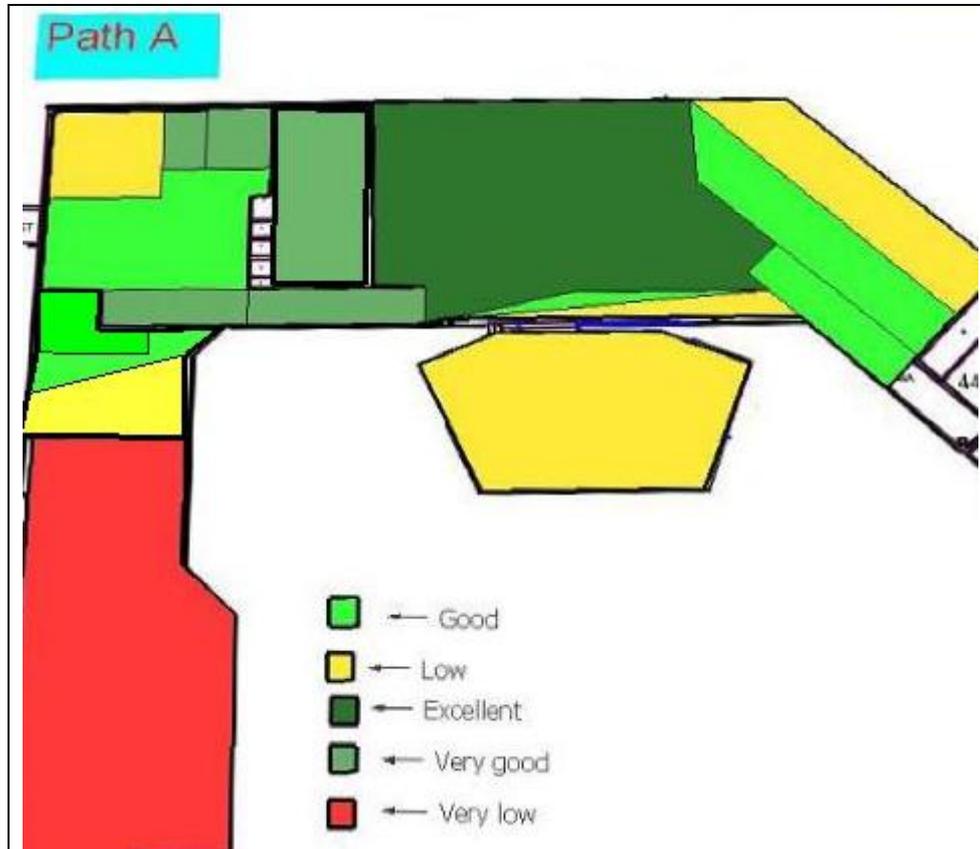
Figure 22. Signal distribution of access point 2 in the path A.

It is seen that signal strength at the positions very close to the AP2 is very strong and has the excellent quality. As the distance between the measuring position and AP2 increases, the signal is attenuated. However, the signal is attenuated at different rates in different directions.

For example, the signal is gradually attenuated on the line-of-sight from AP2 to the room 4408 but the room 4501 is quite close to the AP2, but the signal is greatly attenuated because there are several walls absorbing the signal wave.

3.3.2 Signal distribution of multiple access points at a static position

A small experiment was conducted to investigate the variance of the RSSI received by a device in the experimental environment (Figure 21), which can be seen in Figure 23.

We chose a certain position, point 1, as indicated in Figure 21, to study the signal distribution from multiple access points AP1 and AP2. For each access point 50 samples with a 5-second interval were taken.
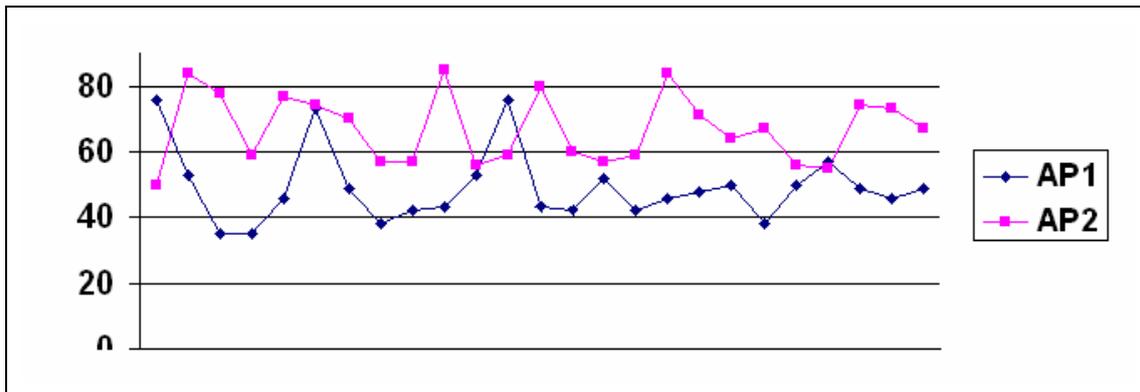
Figure 23. Temporal variance of RSSI at point 1.

As shown, the gathered signal from different access points exhibits different characteristics. Since point 1 is close is very close to AP1, the signal strength from AP1 is rather higher.

Since many factors effect signal propagation (number of people in environment, doors opening and closing, etc), there are some cases in which the signal can be greatly attenuated even if the measuring position is quite close to the access point. The weaker samples reflect this phenomenon.

3.3.3 Effect of user's direction

Because the resonance frequency of water is at 2.4 GHz and the human's body consists of 70% water, the signal is absorbed when the user obstructs the signal path causing extra attenuation [51].

Two sets of measurements, one at the point 1, and one at the point 2 inside the room 4434 were performed to study the effect of user's orientation on signals from AP1.

The measurement was done with four orientations (facing North, West, South, and East of the building) for a period of 2 minutes. The result of the statistics of the RSSI values from the AP1 at point 1 and in the room 4434 is shown in Table 4 and Table 5.

Table 4. Effect of user's direction on RSSI at point 1.

Direction	South	East	West	North
Average (dB)	- 66.68	- 69.68	-78.72	- 68.24

Table 5. Effect of user's direction on RSSI at point 2.

Direction	South	East	West	North
Average(dB)	- 47.02	- 42.9	- 40.77	- 47.42

As seen in Table 4. Effect of user's direction on RSSI at point 1. the RSSI is lowest when the user was facing west and the AP was behind the user, the sample mean was lower at -78.72 dB compared to the highest RSSI of -66.68 dB when the user was facing south. The result shows that the signal strength can be attenuated by 12.4 dB in our case due to the user obstructing the signal from the AP with their body.

As Table 5 shows the signal has been attenuated by 6.65 dB between the highest and lowest signal strength at point 2, but signal quality falls in the excellent interval in all directions.

3.3.4 Long term RSSI variations

To study the long term characteristics of signal propagation, 4000 samples from AP1 at the point 1 was taken with people walking around in corridors and rooms. The same cluster of very good signal strength is observed (-40 to -50 dB). Figure 24 shows the portability distribution of signal strength from AP1 at point 1. As shown, the long term characteristics of signal propagation are very similar to the short term characteristics shown in the Figure 23, except that there is a heavy tail in the long term figure. Since more samples are taken in the long term analysis, a greater number of weaker samples are detected in the collection procedure, which results in a heavy tail.
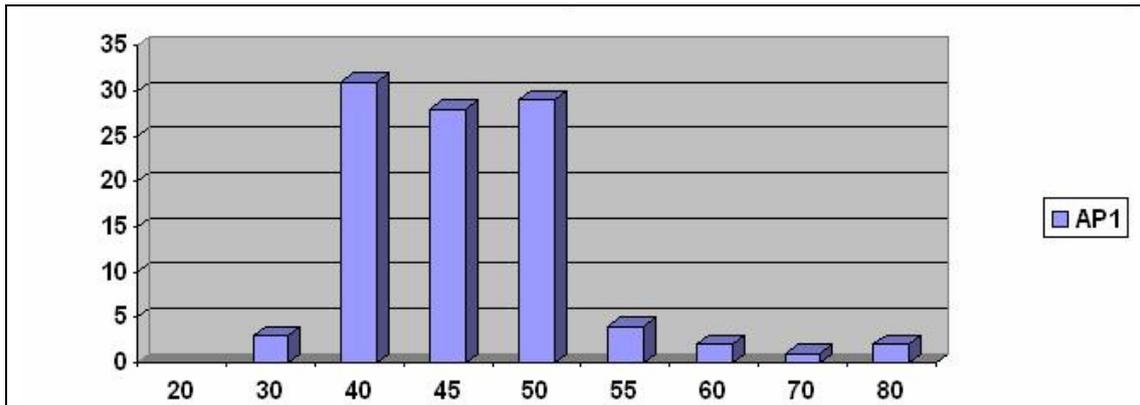
Figure 24. Long term characteristics of AP1 at point 1.

3.3.5 Time dependency of RSSI

Three sets of measurement at point 1 (see Figure 21) at different times were performed to analyze effect of time on the signal strength of AP2. The collected data is shown in Figure 25.

The summary of statistics in Table 6 shows that RSSI varies in different times of the day due to the dynamics of the indoor environment.
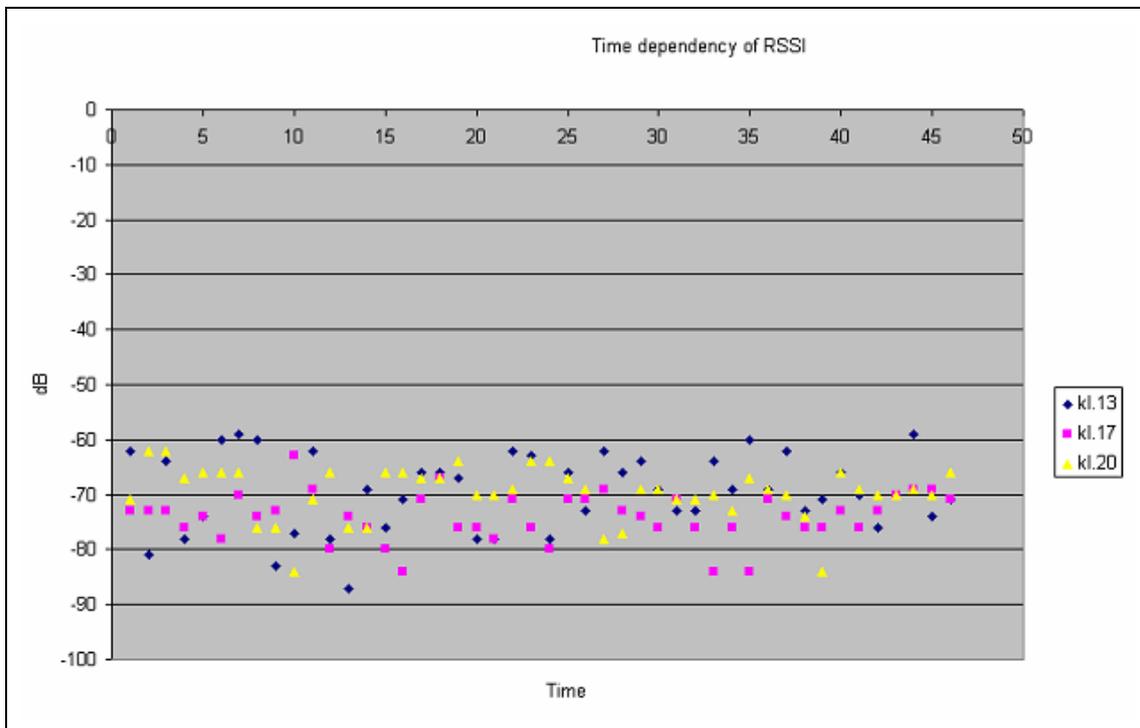
Figure 25. RSSI at different times

As seen in the table, at 17.00 the RSSI was lower than the other times and it can depend on the higher traffic in corridors in that time and because more users were using this access point in that time.

Table 6. Average RSSI at different times of the day.

Time	13.00	17.00	20.00
Average (dB)	-69.54	-74.78	-69.78
Deviation	6.98	4.36	4.92

4. Visualization of collected data

To show the collected data and the generated coverage on the map two different methods using Google SketchUp and SVG (See section 2.7) maps were examined.

4.1 Method 1 using SVG maps

I show the location of access points and the points where RSSI were measured on the SVG map by putting a rectangle at those points. Red rectangles indicate the position of access points and blue is the points where signal strength was measured.

Figure 26 is the map of second floor of Wireless@KTH. As seen four access points are installed in this floor.

The collected data at each point was integrated with that points using Javascript. When clicking on the points, the user will be prompted by a window which shows the collected data.
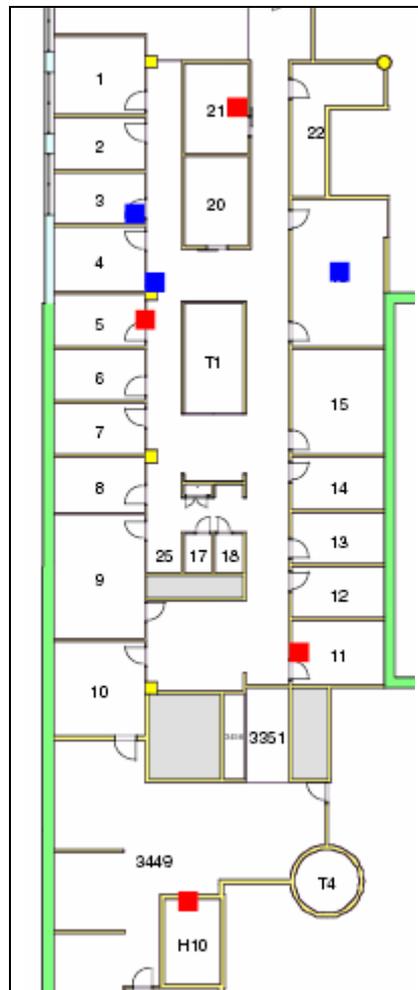
Figure 26. Blue rectangles are places where Open network were scanned and red rectangles are place of APs.

When you click on the red rectangle in the room 21 you will be prompt by a window which shows you the MAC address of this access point. See Figure 27. User will be alerted when clicking on the rectangles.. And when you click on the blue

rectangle in the middle of the room 21, you will be alert by a window which shows a list of collected data. See Figure 28.
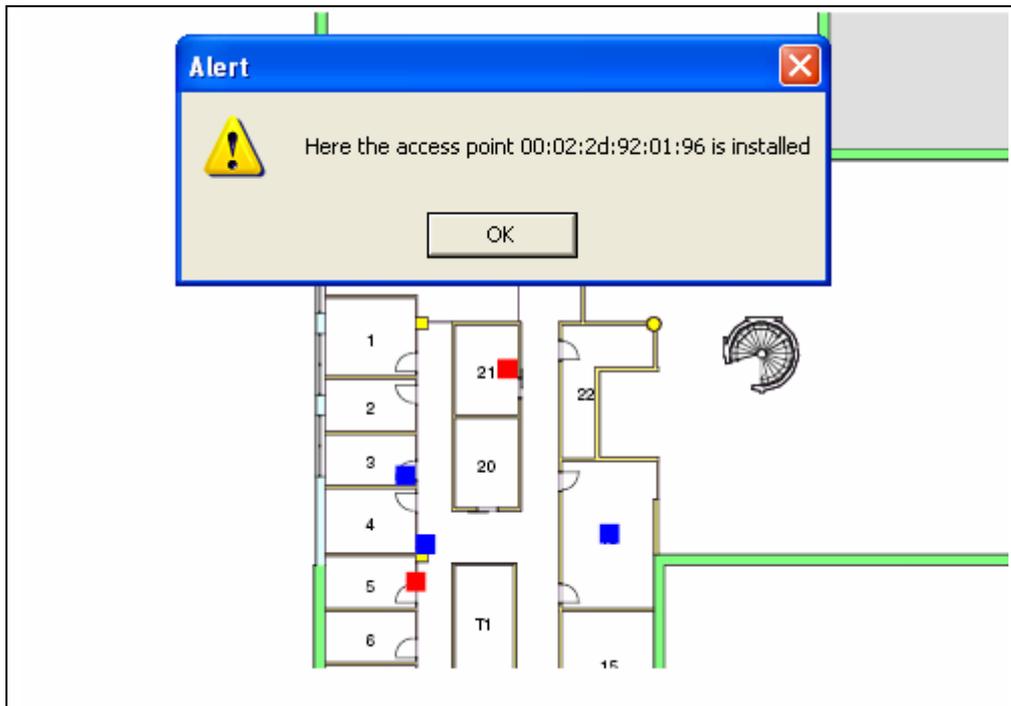
Figure 27. User will be alerted when clicking on the rectangles.
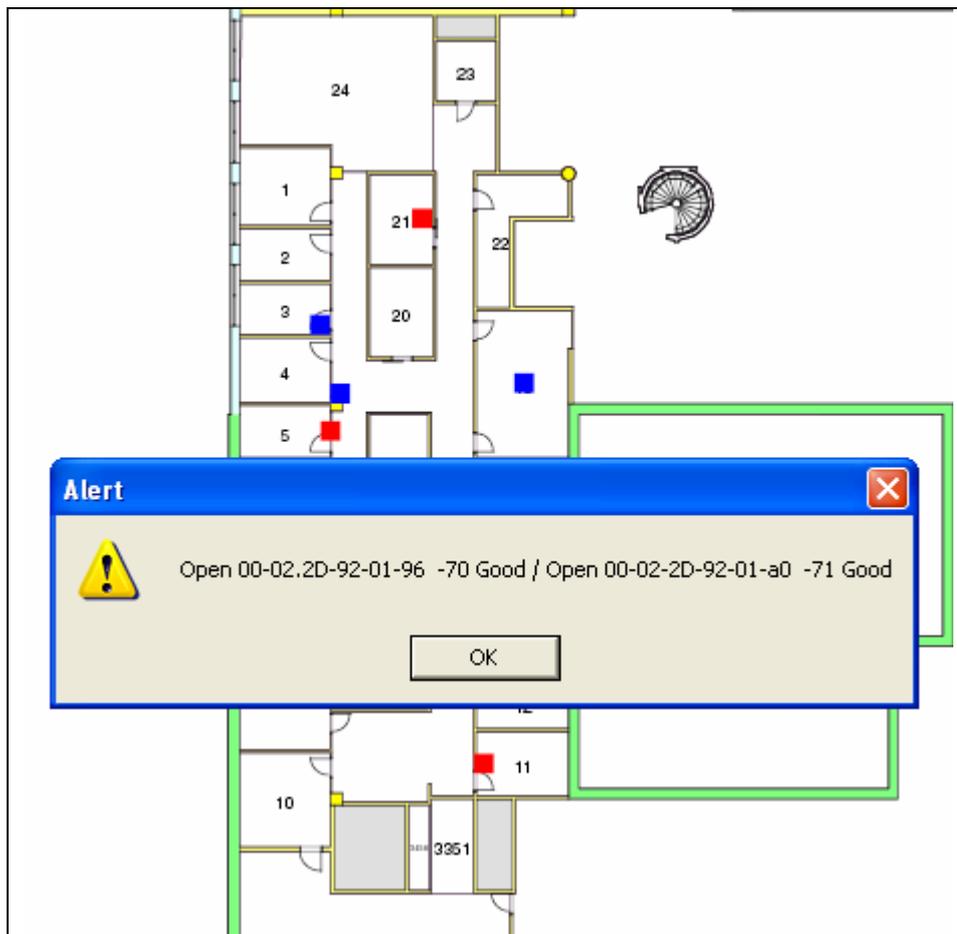
Figure 28. Collected data shown after clicking on the position.

4.2 Method 2 Using SketchUp and Ruby

A Ruby plug-in (See section 2.7.4) was made, which makes it possible to associate links and files with objects in SketchUp. In this way, the collected data was integrated with the objects that show the points of scanning.

First a 3D model of the Wireless@KTH research center was made by Google SketchUp as you see in the Figure 29. Transparent green walls indicate excellent coverage inside the building. X:s are some of points were signal strength has been measured.
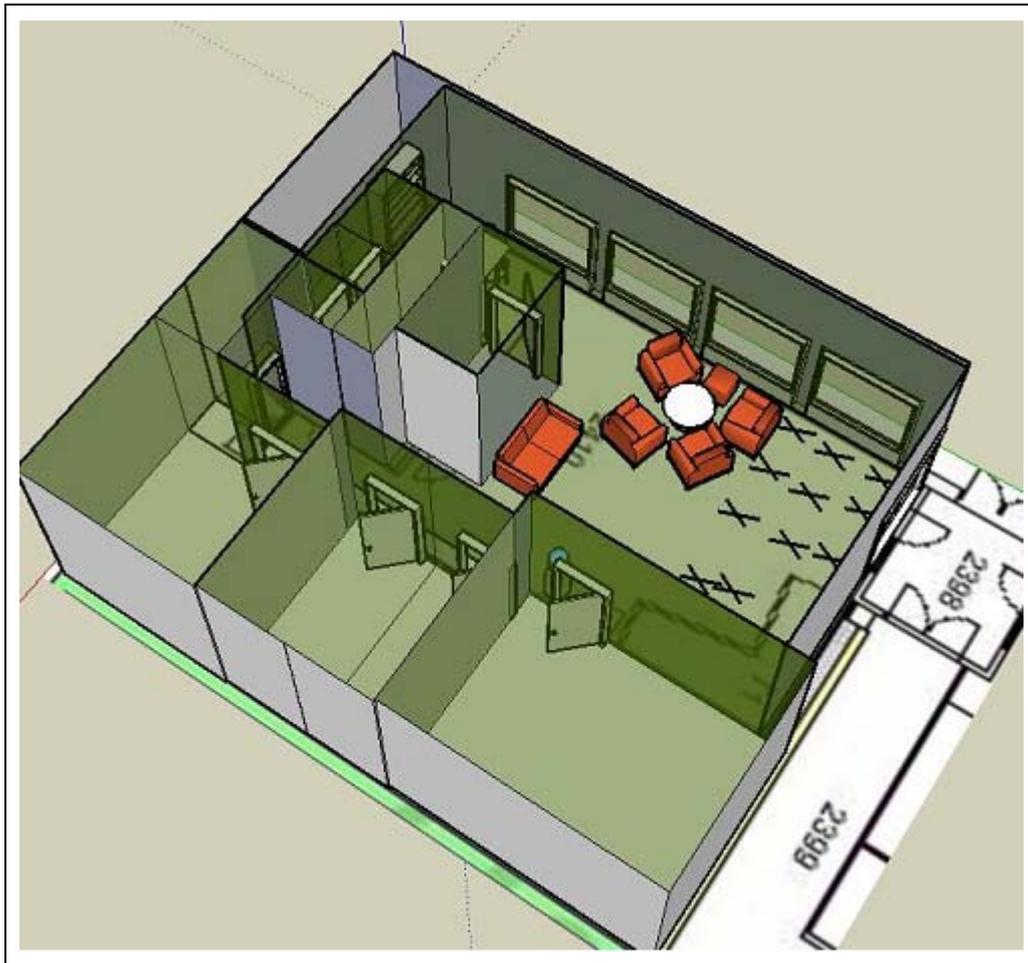
Figure 29. 3D model of Wireless@KTH.

By right clicking on X:s, you can integrate a file or URL to that point. The associated file could be the scanned data or you can upload that file into a website and integrate its URL with that point. See Figure 30 and Figure 31.
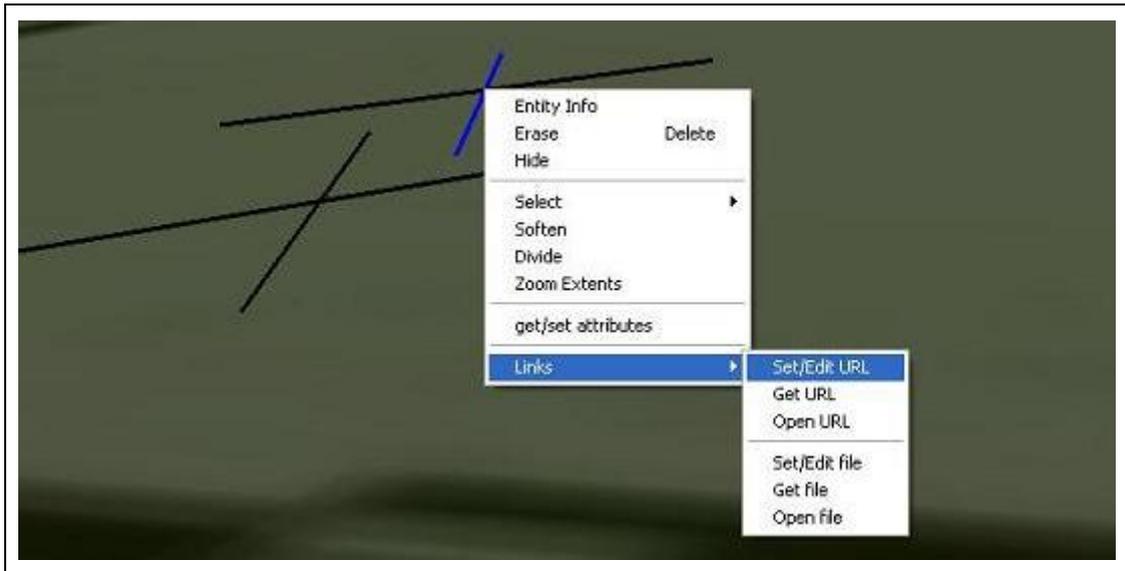
Figure 30. Associating a file with an object.
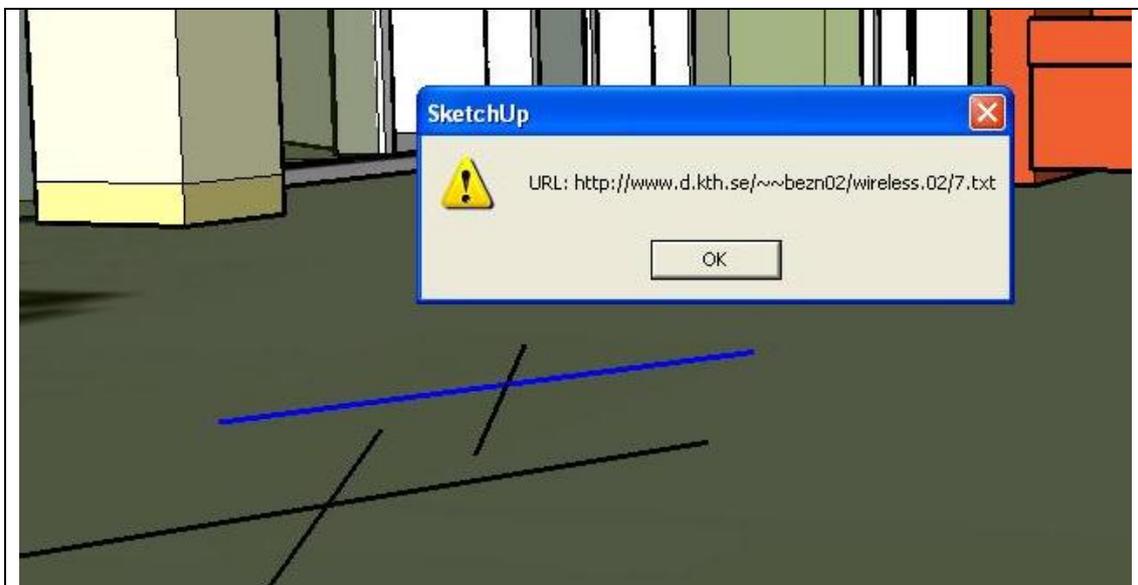
Figure 31. Getting URL.

The 3D model of this building and its associated meta data can be put into the Google Earth for WLAN users and administrators interesting to know the exact extension of the Open wireless LAN.

5. Conclusion

5.1 Discussion

5.1.1 Site Surveying or radio prediction?

In this thesis we used site surveying as a method for measuring the actual real network performance and using this information for building the 3D visualization model. A question maybe arised that why we did not use a planner software as some vendors use for deployment and optimization of the WLANs.

In this work we were not interested in such maps and models, because the propagation models do not work well unless the details of the materials of all the walls, floors, etc are known. As soon as people are presented in the room these tools do not work well - since the presence of people will change the propagation. Even worse, if the people move then all the modeling results are wrong.

Our goal was to build a detailed 3D coverage model integrated with real data which mobile users could use while navigating inside the model to get a better knowledge about the indoor coverage and find their ways to "better" locations, make better decisions, and optimize the travel time and their data transmission.

5.1.2 Access point placement

When designing wireless communication systems, it is very important to know the optimum numbers and locations for the access points. The correct placement of APs is essential. If they are placed too far from each other coverage gaps will be generated, but if they are too close to each other, this will lead to excessive co-channel interferences.

There are two approaches for deployment of an 802.11b WLAN, the designed approach which uses site surveying or radio propagation prediction tools and then utilizes the information gathered by this tool to optimize the number and place of the access points and the Ad Hoc approach [52] which just estimates the number of access points required for covering the area. According to [52] the optimized approach provides higher throughput and less RF interference.

5.1.3 Optimizing individual client access

Due to the 802.11 protocol implementation, access points work as stand-alone devices without a whole view about the whole network. They send signals and accept request for connections from any client and have the capacity to handle 2,408 MAC addresses and clients compete for access on a first-come first-served basis.

In current WLANs, each user scans the available channels to detect its nearby APs and associate itself with the one providing highest signal while ignoring the load on that access point. Because mobile users are not usually distributed uniformly, most of the users can be associated with a few access points while some neighbor access point maybe idle or carry light load. Because the access point is passive in this procedure, performance is degraded as the number of users increases on a specific access point and a client will hang on an access point far too long, causing lower throughput and poorer performance in the WLAN.

Letting the users make more intelligent decisions by providing users with information such as load on every access point and other access point performance metrics would optimize individual client access and maximize the overall system throughput by sending users to the less loaded access points.

Some research about dynamic load balancing has been done [53, 54].

5.2 Future work

Future could consist of automating the collection of data and how to automate the presentation of the resulting experimental data. Further the scanner program could be more users-friendly. Another option would be to integrate the scanned data as XML files to SketchUP directly by a Ruby plug-in.

References

[1]   IEEE 802.11 Standard.
        http://www.ieee.org/portal/pages/about/802std/index.html
        Last access: 2007/03/02

[2]   Wei-Meng Lee, Wireless Surveying on the Pocket PC. 2004/05/27
        http://www.oreillynet.com/pub/a/wireless/2004/05/27/wirelessonPocketPC.html
        Last access: 2007/03/02

[3]   WiFiFoFum,
        http://www.aspecto-software.com/rw/applications/wififofum/index.html
        Last access: 2007/03/02.

[4]   NetStumbler, www.netstumbler.com
        Last access: 2007/03/02.

[5]   MiniStumbler, http://www.netstumbler.com/downloads/
        Last access: 2007/03/02.

[6]   PocketWarrior, http://pocketwarrior.sourceforge.net/
        Last access: 2007/03/02.

[7]   Maxime Flament, and Mattias Unbehaun. Impact of shadow fading in a
        mm-wave band wireless network. Proc. of the 3rd International Symposium on
        Wireless Personal Multimedia Communications, 1, 427-432, 2004.

[8]   Cheverst, K., Davies, N., Mitchell, K., and Friday, A. Experiences of
        developing and deploying a context-aware tourist guide: the guide project.
        International Conference on Mobile Computing and Networking, Boston,
        Massachusetts, United States, 2000.

[9]   http://mappoint.msn.com/(yywddn45s511o0ajdtyvwv55)/Home.aspx
        Last access: 2007/03/02.

[10]   http://www.streetatlausa.com/
        Last access: 2007/03/02.

[11]   http://www.dodgeball.com/
        Last access: 2007/03/02.

[12]   Juuso Pesola, and Sami Pönkänen. Location-aided Handover in Heterogeneous
        Wireless Networks. Wireless Personal Communications: An International
        Journal, Volume 30 , Issue 2-4, 2004.

[13]   Wei Li. Towards a person-centric context aware system. Licentiate thesis,
        March 2006.

[14]   Suvi Ahonen, AND Heikki Laitinen. Database correlation method for UMTS
        location. Vehicular Technology Conference, 2003. VTC 2003-Spring. The
        57th IEEE Semiannual.

[15]   P. Bahl, and V. Padmanabhan,. Radar: An in-building RF-based user location
        and tracking system. INFOCOM 2000. Nineteenth Annual Joint Conference of
        the IEEE Computer and Communications Societies. Proceedings. IEEE, 2000.

[16]   Blake M. Harris. Amulet: Approximate Mobile User Location Tracking
        System. University of Rochester.

[17] M.Youssef and A. Agarawala, The Horus WLAN location determination system. In MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services.

[18] Marion Hermersdorf. Indoor positioning with a WLAN access point list on a mobile device. WSW'06, October 31, 2006, Boulder, CO, USA.

[19] L. Aalto, N. Gothlin, J. Korhonen, and T. Ojala, Bluetooth and WAP push based location-aware mobile advertising system. Proceedings of the 2nd international conference on Mobile systems, applications, and services, Boston USA, 2004.

[20] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/snmp.htm Last access: 2007/03/02.

[21] Ekahau, Inc., Ekahau Technology and Products. http://www.vtt.fi/virtual/navi/expo2003/Ekahau030402.pdf

[22] John Krumm, and Eric Horvitz. Locadio: Inferring Motion and location from WiFi signal strength. MobiQuitous 2004, Boston USA, 2004.

[23] INTEL RESEARCH SEATTLE. PlaceLab: A Privacy Observant Location System, http://www.placelab.org. Last access: 2007/03/02.

[24] Wifi Map, Silver Lake Ohio, USA. http://www.pisarsky.com/local/maps/wifislak.html Last access: 2007/03/02.

[25] UCSB campus modeling and visualization. http://ilab.cs.ucsb.edu/projects/lihua/lihua.html Last access: 2007/03/02.

[26] iSPOTS. http://ispots.mit.edu/ispots.html Last access: 2007/03/02.

[27]   http://en.wikipedia.org/wiki/Augmented_reality Last access: 2007/03/02

[28] http://www.fileinfo.net/extension/vrml Last access: 2007/03/02.

[29] Algorithm & Applications Group, Campus Navigator. http://parasol.tamu.edu/groups/amatogroup/research/campus-nav/ Last access: 2007/03/02.

[30] Google Maps API. http://www.google.com/apis/maps/ Last access: 2007/03/02.

[31] ASP.net AJAX. http://ajax.asp.net/ Last access: 2007/03/02.

[32] Google Maps. http://maps.google.com/ Last access: 2007/03/02.

[33] Campus Navigator, http://uwcampusnav.sourceforge.net/project.html Last access: 2007/03/02.

[34] Java Wireless Research API, http://media.csee.ltu.se/~johank/javawrapi/ Last access: 2007/03/02.

[35]   USCD ActiveCampus. http://activecampus.ucsd.edu/
        Last access: 2007/03/02.

[36]   Arvinder Singh, and Ali Taheri. LOCUS: Wireless LAN location sensing.
        http://web.cs.wpi.edu/~emmanuel/MQPs/locus/
        last access: 2007/03/12.

[37]   BATIK API, Vincent Hardy, and Thierry Kormann. Leveraging SVG on the
        JAVA platform with Batik.
        http://www.gca.org/papers/xmleurope2001/papers/html/s21-1.html
        Last access: 2007/03/12.

[38]   Batik API Specification. http://www.stylusstudio.com/api/batik/
        Last access: 2007/03/02.

[39]   Batik JAVA SVG Toolkit. http://xmlgraphics.apache.org/batik/
        Last access: 2007/03/02.

[40]   Peter Steenkiste. Aura: Invisible ubiquitous computing. Project Aura
        http://www.cs.cmu.edu/~aura/
        Last access: 2007/03/12.

[41]   Windows Mobile.
        http://www.microsoft.com/sverige/windowsmobile/default.mspx
        Last access: 2007/03/02.

[42]   Symbian OS. http://www.symbian.com/
        Last access: 2007/03/02.

[43]   Palm OS. http://www.palm.com/
        Last access: 2007/03/02

[44]   John Ioannidis, Dan Duchamp, and G.Q. Maguire Jr. IP-based Protocols for
        Mobile Internetworking. SIGCOMM'91 Conference: Communications
        Architectures and Protocols, pages 235-245. Association for Computing
        Machinery, September, 1991.

[45]   Adobe SVG Viewer. www.adobe.com/svg/viewer/install/
        Last access: 2007/03/02.

[46]   Jorge Gustavo Rocho. A Java based SVG viewer for the Pocket PC.
        http://www.svgopen.org/2003/paperAbstracts/JavaSVGViewer4PocketPC.html
        Last access: 2007/03/02.

[47]   Daniel Bogaard, Ronald P.Vullo, and Christopher D. Cascioli. Better HTML
        Web: Dynamically Generated SVG web sites. Rose-Hulman Institute of
        Technology, Terre Haute, Indiana, USA.

[48]   Microsoft SQL server 2005 Mobile.
        http://www.microsoft.com/sql/editions/sqlmobile/sqlmobileresources.mspx
        Last access: 2007/03/02.

[49]   Google Earth, http://earth.google.com/
        Last access: 200703/02.

[50]   Google SketchUp. http://sketchup.google.com/
        Last access: 2007/03/02.

[51]   Andrew. M. Ladd, Kostas E. Bekris, and Algis Rudys. Robotics-based location sensing using wireless Ethernet. Proceedings of the 8th annual international conference on Mobile computing and networking, Atlanta, USA, 2002.

[52]   Jim Krupka. IEEE 802.11b wireless LAN: Engineered Design Versus Ad Hoc Approach. http://www.bechteltelecoms.com/docs/bttj_v1/Article13.pdf Last access: 2007/03/12.

[53]   Yigal Bejerano, Seung-Jae Han, and Li Bell. Fairness and load balancing in wireless LANs using association control. Proceedings of the 10th annual international conference on Mobile computing and networking, Philadelphia, USA, 2004.

[54]   Shiann-Tsong Sheu and Chih-Chiang Wu, Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 Wireless LAN, *Tamkang Journal of Science and Engineering, vol 2, No 1, pp.45-52 (1999).*

[55]   Cisco Wireless Location Appliance, http://www.cisco.com/en/US/products/ps6386/ Last access: 2007/03/02.

[56]   John C.Stein. Indoor Radio WLAN Performance, part 2: range performance in a dense office environment. Intersil Corporation, Palm Bay, Florida.

[57]   www.wifimaps.com Last access: 2007/03/04

[58]   www.nodedb.com Last access: 2007/03/04

[59]   Chris Lentz. 802.11b wireless network visualization and radio wave propagation modeling.

[60]   CommView for WiFi. http://www.tamos.com/products/commwifi/ Last access 2007/03/04

[61]   CommView for PPC. http://www.tamos.com/products/commwifippc/ Last access 2007/03/04

[62]   Kismet. http://www.kismetwireless.net/ Last access 2007/03/04

[63]   WirelessMon 2.0. http://www.passmark.com/products/wirelessmonitor.htm Last access: 2007/03/04

[64]   ManageEngine WiFi Manager 4.5.0. www.manageengine.adventnet.com/products/wifi-manager/index.html Last access: 2007/03/04

[65]   Herecast WiFi scanner. www.herecast.com Last access: 2007/03/04

[66]   Johanna Janse van Rensburg and Barry Irwin. Wireless network visualization using radio propagation modeling.

[67]   Herecast: WiFi location based services/802.11 positioning system. http://www.herecast.com/ Last access: 2007/03/08

[68]   Ruby programming language.
         http://www.ruby-doc.org/docs/UsersGuide/rg/whatisruby.html
         Last access: 2007/03/08