2007-11

Wireless Positioning Techniques: a Developer's Update

Seamus Rooney
Dublin Institute of Technology, seamus.rooney@gmail.com

Keith Gardiner
Dublin Institute of Technology, keith.gardiner@dit.ie

James Carswell
Dublin Institute of Technology, jcarswell@dit.ie

Recommended Citation

Rooney, S., Gardiner, K. & Carswell, J. (2007) Wireless positioning techniques: a Developer's update. *7th International Symposium, W2GIS 2007.* J.M. Ware and G.E. Taylor (Eds.): W2GIS 2007, LNCS 4857, pp. 162–174.

2007-11-01

Wireless positioning techniques: a Developer's update

Seamus Rooney
Dublin Institute of Technology, seamus.rooney@gmail.com

Keith Gardiner
Dublin Institute of Technology, keith.gardiner@dit.ie

James D. Carswell
Dublin Institute of Technology, jcarswell@dit.ie

Wireless Positioning Techniques: A Developers Update

S. Rooney *, K. Gardiner , J.D. Carswell

Digital Media Centre, Dublin Institute of Technology, Aungier St., Ireland
{seamus.a.rooney, keith.gardiner, jcarswell}@dit.ie

Abstract. This paper describes the current efforts to develop an open source, privacy sensitive, location determination software component for mobile devices. Currently in mobile computing, the ability of a mobile device to determine its own location is becoming increasingly desirable as the usefulness of such a feature enhances many commercial applications. There have been numerous attempts to achieve this from both the network positioning perspective and also from the wireless beacon angle not to mention the integration of GPS into mobile devices. There are two important aspects to consider when using such a system which are privacy and cost. This paper describes the development of a software component that is sensitive to these issues. The ICiNG Location Client (ILC) is based on some pioneering work carried out by the Place Lab Project at Intel. (Hightower et al., 2006) The ILC advances this research to make it available on mobile devices and attempts to integrate GSM, WiFi, Bluetooth and GPS positioning into one positioning module. An outline of the ILC's design is given and some of the obstacles encountered during its development are described.

Keywords: Location based services, Location prediction, WiFi positioning

1 Introduction

The ICiNG Location Client (ILC) is a part of an overall project called ICiNG. The ICiNG project is about researching a multi-modal, multi-access concept of e-Government (DIT, 2006). It develops the notion of a 'Thin-Skinned City' that is sensitive to both the citizen and the environment through the use of mobile devices, universal access gateways, social software and environmental sensors. Intelligent infrastructure enables a Public Administration Services layer and a Communities Layer.

The ICiNG project will set up test-beds in high-profile European locations such as Dublin, Barcelona and Helsinki to act as 'City Laboratories' for researching, evaluating and demonstrating technologies and services using intelligence in the environment.

In the ICiNG project, the fundamental requirement for this type of service interaction is location. The ILC software component is about researching and developing an open source standalone location determination application that will be able to use a hybrid of different technologies to determine its location. For example, the ILC in any given environment will be able to return a location providing there is at least one of the beacon technologies available and these include GPS, WiFi, GSM and Bluetooth. The ILC has being developed using a modularized component architecture so that each scanning component can and does run independently of the other technologies being available. This approach also has the advantage of being able to make any changes that need to be made with as little disruption as possible. This makes it easier to update specific modules and even add or remove modules without affecting the overall performance of the ILC. The ILC will not communicate with any external applications that are not installed on the device. Instead, it uses a Record Management Store (RMS) to store a limited amount of information and only at the users consent. The ILC is by design an open source, network independent, location determination mobile application that can utilise GPS, WiFi, GSM and Bluetooth information or any combination of them, to calculate location. (Kilfeather et al., 2007)The paper is organised as follows: Section 2 gives a brief summary of the related work carried out by the Place Lab research group. Section 3 gives a brief description of the ICiNG test beds, privacy considerations, location considerations and introduces some specific terminology. It then gives a more detailed description of the ILC architecture and its various modules and finishes by highlighting various development issues encountered so far. Finally, section 4 gives the conclusions for the project to date and comments on the expectations for the ILC's location determination methodology.

2 Related Work

2.1 Place Lab

Place Lab was an Intel research project that ended in 2006. The goal of Place Lab was to try and exploit freely available radio signals (GSM, WiFi, and Bluetooth) for location determination of a user's HHD (Hand Held Device). They did this by building a database of the known transmitting locations (beacons) of these signals, and then used the *assumed distances* (e.g. based on signal strength) from the user HHD to these radio beacon locations in a trilateration[1] procedure to determine the user's location. (Without initial

[1] Trilateration is a method of determining the relative positions of objects using the geometry of triangles in a similar fashion as triangulation. Unlike triangulation, which uses angle measurements (together with at least one known distance) to calculate the subject's location, trilateration uses the known locations of two or more reference points, and the measured distance between the subject and each reference point. To accurately and uniquely determine the relative location of a point on a 2D plane using trilateration alone, generally at least 3 reference points are needed.

knowledge of the direction to each of the beacons, a triangulation procedure using angles is not possible.) However, at the time, certain technologies were not yet advanced enough to take Place Lab fully mobile as some of the signal spotters worked on mobile phones (i.e. Bluetooth) while others needed a laptop (i.e. WiFi).

In the case of the ILC, the technologies that prevented Place Lab from bringing this technology fully mobile have advanced to the stage where mobile devices now offer integrated WiFi, GPS and Bluetooth capabilities. This advancement of these technologies is the basis for the ILC's design and like Place Lab, the ILC is designed to determine the position of a mobile device using passive monitoring that gives the user control over when their location is disclosed, laying the foundation for privacy-observant location-based applications. (LaMarca et al., 2005)

2.2 Location Calculations

The ILC will implement and test several different location calculation algorithms to determine the best possible location. After initial testing we will look at improving upon one or more of these algorithms while continuously testing for the most accurate results. There may be a performance sacrifice on a mobile device to achieve increased accuracy or there may be improvements to algorithms that could deem one algorithm more suitable to a mobile device than another. Only after some careful testing will it be possible to make these assumptions. A description of the algorithms that we will implement and test initially is given in the following.

2.2.1 Centroid Location Determination

This is one of the simplest algorithms that will be used by the ILC. It involves taking into account the locations of all known beacons in the area and then positioning the user mathematically at the centre of them. This approach ignores many things that could improve the location determination including beacon signal strength, confidence in the beacons location and environmental issues (i.e. tall buildings, hills, buses, etc.). (Hightower et al., 2006)

2.2.2 Weighted Centroid Location Determination

This is very similar to the standard Centroid algorithm but it takes into account other values in the RMS database when calculating a location. For example, the signal strength of each beacon can be taken into account to further determine if the ILC is nearer one beacon or another. Also, as previously stated, there will be a test bed used to test the ILC that will have fixed beacons where the exact position of each beacon is known. Initially, we will only be using these beacons but later during testing, we will carry out some Wardriving in and around the test bed area and add unknown beacons to the RMS database that have estimated locations. In this case, the beacons we know the exact locations of will get a greater weight when calculating location, than beacons that are detected using Wardriving.

2.2.3 Assisted ILC

Much like assisted GPS, assisted ILC will use an Assistance Server but on the phone. For example, if a mobile device was within signal distance of a known Bluetooth beacon and then moves out of range of that Bluetooth beacon this information would not be discarded straight away. Assisted ILC will use this information to help correct the location determination for a period of time. As time passes the weight of the correction will also decrease. Also depending on the type of beacon being used for the correction, there will be a weight attached to the expected accuracy it is expected to correct for.

Initially we have decided to implement these three algorithms and test them out in our test bed environment to determine which one offers the best overall performance in computational speed, resource efficiency and energy consumption and at the moment we are currently only testing the first algorithm “Centroid” with the first prototype ILC. There are other techniques that were considered, e.g., Particle Filters (Hightower and Borriello, 2004) and Fingerprinting, which are used in RADAR (Bahl and Padmanabhan, 2000) but initially it was decided that these might be too costly in terms of performance on a mobile device. Another possible technique is Gaussian Processes for Signal Strength-Based Location Estimation (Ferris et al., 2006) which seems more optimal for mobile devices, but initially we aim to test the first three Location Determination Algorithms deciding later if a different approach is required.

3 ICiNG

3.1 ICiNG Test Bed

The general approach of the Dublin test bed is to bring existing wireless and wired infrastructure belonging to the Dublin Institute of Technology and Dublin City Council together within an experimental on-street wireless network (Figure 1). Access to the new wireless network will, in the first instance, be open, free and supported by the Dublin ICiNG project team.

The ICiNG street signs will be the WiFi/Bluetooth Access points and the ILC will use its internal logic to determine a best guess location from the known positions of the street signs and other radio signals in the area.

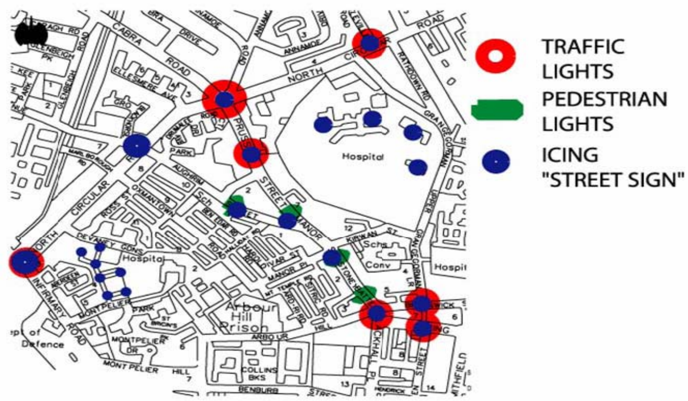

Fig. 1. Grangegorman Map: The Dublin Test-bed for the ICiNG System with the wireless access points highlighted.

3.2 Specific Terminology

The *Icing Mobile Client (IMC)* refers to the complete set of application components on the mobile device. The IMC is comprised of:

- The *ICING Location Client (ILC)* whose purpose is to calculate and make available the device location to the MDA based on GPS, Bluetooth, and WiFi Beacon information.
- A number of *Mobile Device Applications (MDA)*. An example of an MDA is an accessibility application that enables users of the ICiNG system to report accessibility issues to the City Council using a Jabber client[2] extension.

3.3 Location Considerations

There are many Location Based Services (LBS) identified in the ICiNG project. These range from providing a location tracking sensor network to retrieving metadata based on a mobile device's location. While these services are heterogeneous in nature they all require a method of determining the location of a particular device or sensor. There are many existing systems available to provide this location information, some using cell services provided by mobile telecoms providers or others using satellite technology such as GPS.

[2] An open, secure, ad-free alternative to consumer IM services like AIM, ICQ, MSN, and Yahoo. Jabber is a set of streaming XML protocols and technologies that enable any two entities on a network to exchange messages, presence, and other structured information in close to real time.

However, as discussed previously each of these technologies and services have inherent advantages and disadvantages. Some services operate well in urban areas and in areas of high cellular radio density while others perform well where line of sight to satellites in the GPS system is established. Also, beyond the purely technical or technological considerations to be taken into account in location determination are issues of privacy and safety which location technologies raise. (Vossiek et al., 2003)

Fortunately, the issue of deciding which of these technologies to use is being somewhat mooted by the increasing trend of mobile devices to incorporate multiple access technologies in the same platform. The availability of GSM, WiFi, Bluetooth and GPS on the same device offers the possibility of intelligently using all these technologies in combination to improve location availability and accuracy.

3.4 Privacy & Cost

One of the main concerns during the ILC design phase was that of user privacy. We did not want to design a system that would or could be used to track a citizen's location without their knowledge or prior consent. Instead, we wanted to develop a system where all the location determination could be done on the mobile device itself and then, only if the user wanted, it would be possible to inform the rest of the ICiNG system of their location. In this way, the user will have full control over their location and would not have any concerns about their movements being tracked without their knowledge.

For users that allow the ILC to disclose their location to ICiNG services, we also wanted a system that would address any other privacy concerns they might have. Of these issues, we identified the following to be particularly important; location information retention, location information use, and location information disclosure.

For ICiNG, it was decided at an early stage that location information retention, if it needed to happen at all, would be only for a short task-specific time frame depending on whether the user was partaking in specific studies or if they had signed up to a service that required their movements to be tracked. For example, a parent could register their child's mobile phone to such a service to monitor their child's whereabouts. Another issue that needed to be addressed was that of disclosing movements of users to 3rd parties, which the ILC never directly does.

Although we have addressed privacy as a consideration, the very nature of the ILC is that for the location information to be useful, a 3rd party application needs access to it - and the ILC has no control over what 3rd party applications do with this information. However, 3rd party applications do need to access the record store that holds the devices location and this can and will be restricted to verified applications. Unfortunately we are not able to enforce our policies onto other applications and acknowledge that providing such a facility does require the 3rd party application to address privacy in their own design. Although we have no control over what happens the information once it is used by another application a system such as Casper (Mokbel et al., 2007) should be considered if there is any communication with a ermote server to provide a location based service.

The ILC was designed to be a monetarily zero cost solution to location determination. The only cost we see for the user is the time cost to install the application and the bandwidth cost to receive freely available radio signals. There is also a CPU cost of running the application and after testing the initial prototype, we will endeavor to reduce the overall processing cost of the application using more efficient algorithms, etc.

So far we have discussed some of the different positioning techniques available, and some of their advantages and disadvantages. We identified concerns that most people would have about their privacy being infringed upon and what the possible solutions to these issues are. We also noted, due to technology limitations of the day, what Place Lab was not able to do in bringing a fully mobile location based system to fruition, and how ICiNG would take the next step and extend their work by designing and developing such a system. The next part of this paper gives a more detailed overview of the ICiNG system, focusing on the ILC.

3.5 ILC – ICiNG Location Client

For the ILC, we use a combination of technologies to develop a location determination system that integrates the best features of all technologies available. By using all these technologies together, any disadvantage that each individual technology has diminishes. The ILC is designed as provider-network independent, privacy sensitive and zero cost (in terms of network resource usage) software component that allows mobile devices to determine location by a "best guess" methodology. The prototype ILC is designed to run on a Series 60 (3rd Edition) mobile phone running the Symbian operating system (version 9.x), although other platforms and operating systems could be accommodated with relatively minor changes. Figure 2 shows how this architecture works together.

There is a set of defined rules that the ILC uses when searching for beacons in the RMS Database. It is dictated by the degree of accuracy that should be expected depending on the type of beacon. Although all beacons in the RMS Database are read, it first looks for Bluetooth beacons and if any are found it will discard the any other beacons used for determining the location; hence the degree of accuracy should be <10 meters. Next, if there are no Bluetooth beacons, it looks for a GPS reading. If there is no GPS reading then it looks for WiFi beacons and if there are no WiFi beacons, it looks for GSM beacons. As the technologies used changes, so will the degree of accuracy the tracker is providing. Even though beacons with a smaller weight for accuracy are disregarded in the location determination returned to the MDA, this beacon data does not get totally discarded. If the more accurate beacons become unavailable and the ILC switches to the less accurate beacons in its Database for triangulating position, then the less accurate location gets a correction applied based on its proximity to the last known more accurate beacons. (Rooney et al., 2007)

The above rule set and the respective weights that will be applied to each technology is still being refined and will only become clear after prototype testing with different configurations, a phase that has just commenced. Finer points like the expected accuracy from each technology, how the constant use of each technology effects the battery life on a mobile device and hence how many times each signal should be scanned to get the maximum battery life/accuracy ratio will all be tested and reconfigured in time. The different characteristics each of these technologies brings will determine the most efficient and accurate rule set for combining these technologies to work seamlessly together. Currently, we are only testing using the first basic algorithm and plan on doing extensive testing with it. Once this is completed, and we have data to work with, we will move onto implementing and testing the next two algorithms.

3.5.1 Java Bluetooth Spotter

The Bluetooth spotter polls the Bluetooth Terminal hardware and scans for any Bluetooth devices in range, and any devices found in range are returned to the tracker module.

In the case of the Bluetooth spotter a list of the currently detected devices is returned with some exceptions. Take the case of other mobile phones in the vicinity, for location determination this data is inadequate because mobile devices are not stationary and have no permanent location associated with them. Therefore, any Bluetooth beacons that are classed as mobile are filtered out of the list before being returned to the tracker module.

3.5.2 Java GPS Spotter

The GPS Spotter communicates with a GPS receiver to obtain Lat/Long coordinates and returns these coordinates to the tracker module. In the initial implementation of the ILC an external GPS receiver is used. The GPS spotter is used to improve the accuracy of the ILC when used outdoors as the density of WiFi access point’s naturally decreases.

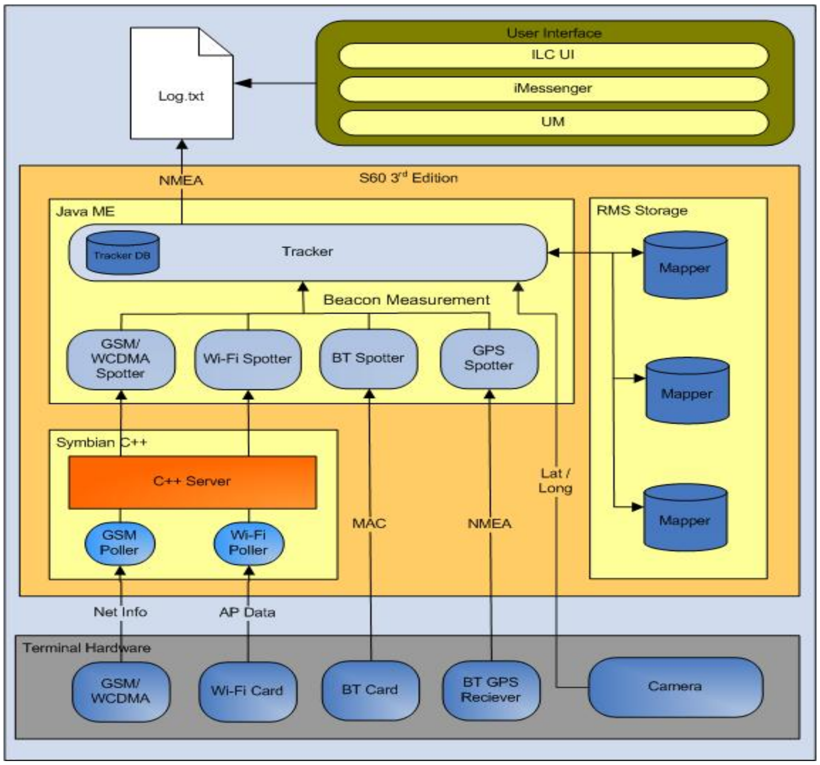

Fig. 2. The architecture of the Icing Location Client (ILC)

3.5.3 Java & C++ GSM Spotter

The Java GSM spotter sends requests to the C++ GSM spotter to retrieve the current GSM tower information and returns it to the tracker module.

In this case, the GSM spotter only returns one set of information at any one time as there is a restriction inherent in the Symbian S60 architecture that prevents any application from obtaining a full list of the currently available masts. This restriction somewhat affects the accuracy of the ILC when in a remote area where the availability of WiFi and Bluetooth is virtually zero as trilateration cannot occur without at least three detectable beacons. Although we are considering different approaches to get around this, none have so far been implemented or tested. For example, a short list of GSM towers that have been scanned in the last 30 seconds and then performing a trilateration of the recorded towers while giving greater weight to the cell towers that were scanned nearer the current time is one such solution.

Upon receiving a request from the Java GSM Spotter, the C++ GSM spotter polls the GSM terminal hardware to determine the cell information about the currently active cell tower and returns this data back to the GSM spotter.

It does this by first opening up a listening port that listens for beacon requests from the Java GSM spotter. Upon receiving a request it polls the mobile device terminal hardware for network information. When the data is returned, the information is collated and sent through the open socket. A flag is then set to notify the server to wait until another request arrives from the Java GSM Spotter.

3.5.4 Java & C++ WiFi Spotter

The Java WiFi spotter sends requests to the C++ WiFi spotter to retrieve information about the WiFi access points present in the area and returns it to the tracker module**.**

The WiFi spotter is similar to the GSM spotter in that it has to poll a native application to obtain the currently detectable access points in the vicinity. It also has the added advantage of being able to return a list of available access points in the vicinity. This makes WiFi a perfect fit for the ILC. With a complete list of access points it is possible to estimate the location using WiFi alone if there are three or more AP’s available.

Upon receiving a request from the Java WiFi spotter, the C++ WiFi Spotter polls the WiFi terminal hardware to determine the MAC addresses of any WiFi access points in range and returns this data to the Java WiFi spotter.

This module first opens up its own port and then waits for a request for information from the Java WiFi spotter. When a request is made to the Spotter, it then has to open a connection with the monitor server (Terminal hardware interface). It subsequently obtains the attributes of the WiFi Access points in range. When this is complete, the information is collated and returned to the Java WiFi spotter

3.5.5 Semacode Spotter

The Semacode spotter module is an additional component of the ILC that enables the user to take a photo of a Semacode tag. Semacode tags are 2D barcode stickers that can be placed at virtually any location, for example, on lampposts. It is possible to encode the barcode with any type of information usually a URL. In the case of the ILC the barcodes are encoded with the location of the barcode. The Semacode spotter is used to record the 2D barcodes and translate them into Lat/Long coordinates. The spotter then returns these accurate coordinates to the tracker module. If the Semacode tags are positioned correctly using highly accurate GPS, the location obtained from using the tags is the most accurate data available to the ILC. Therefore, when a Semacode tag is decoded and a location obtained from it, this position is deemed to be highly accurate and all other estimates are overridden.

3.5.6 Tracker Module

The Tracker module is responsible for collecting beacon information from the four Java spotters and organizing the beacon information into a local tracker database. Using this database of beacons the tracker uses trilateration to determine the current position of the mobile device. It is at this point in the process that other positioning algorithms can be used to try and estimate a more accurate position, some of which will be discussed in section 3.5. The process used by the tracker is as follows,

1. When the ILC starts, the tracker module implements four spotters for Bluetooth, WiFi, GSM and GPS.
2. Upon validation that each hardware component is present, the tracker module polls the four Java spotters for any beacons in range of the phone.
3. The Java spotters then return information about any beacons in range.
4. The tracker module then stores this information in the tracker database.
5. The tracker module then checks to see what beacons are in the tracker database.
6. Depending on the beacons available, the tracker chooses which beacons to use and then compares them against known beacons in the RMS Mapper Database.
7. If the current beacons are unknown, then the tracker module will revisit the tracker database to select the next viable beacons to check against the RMS Mapper Database.
8. Once the tracker module's beacons have been checked against the RMS Mapper Database, it then attempts to trilaterate the best possible position based on the current beacon information.
9. When the position has been determined, it is inserted into the RMS Location Database with the current date and time. This information is then only available to other applications that have been verified as legitimate users of the ILCs location calculation. This verification is done when the application is being packaged.

3.5.7 Tracker Database

The tracker database is a list of the beacons that are currently in range of the mobile device. Each element in the list is compared against the known beacon database, or Mapper database, in order to determine their locations and subsequently trilaterate the devices position.

3.5.8 RMS Mapper Database

The RMS Mapper database is a database of known beacons and their associated locations. This database is used by the Tracker module to lookup beacon locations which are then used to trilaterate the current position of the mobile device. Initially this database only contains data about known beacons and has been manually recorded and is very accurate. In subsequent versions of the ILC it will be possible to use databases created directly from Wardriving and that can be downloaded from websites like [www.wigle.net]. In this scenario, the ILC will be capable of connecting to an online database of known access points and download beacon information for the current location. Also, a more effective possibility will be to use Bluetooth to push local beacon data to mobile devices when they enter certain Bluetooth areas.

3.6 Development Issues

Development for mobile devices and in our experience Symbian C++ and J2ME does not come without its own problems, few of which are documented and only through community forums can a solution be found in most cases. As we have found very little documentation available we decided a section on the pitfalls we have encountered could be helpful to the readers of this paper.

3.6.1 Computer Hardware

One of the main issues we encountered was that the development environment would not work correctly on different types of hardware, mainly on laptops. There is no recommend hardware besides the standard information you get in relation to CPU speed, memory size and HDD size. We encountered numerous intermittent problems while trying to develop on different laptops with different specifications and numerous clean installs of the operating system. In the end, when we switched to a desktop 95% of the intermittent problems we were having were solved. There is also some issue with the integration of the different software packages you need to install. For example, each phone comes with the Nokia PC suite which can be installed and uninstalled with no issues. But if you install the Carbide j SDK and the J2ME Series 60 SDK which is used to develop java applications for Series 60 phones, some problems occur. For Example, if you need to uninstall the Nokia PC suite, the uninstaller also removes various windows drivers.

3.6.2 Security certificates

Developing on the new Symbian Series 60 3rd edition platform has the addition of a new security level to deal with. While developing and creating a new application, the SDK automatically generates what is called a UID for each different application type. When you attempt to test your application on an actual device there are some restrictions. Even though it runs fine in the emulator, it will not install on the device itself. The solution is to go to www.symbiansigned.com, download an application called the DevCertRequest and request a development certificate that includes the capabilities you want for your application. This application will generate a public key for the desired application. Using this key return to the Symbian signed website and upload the key to receive the certificate. On receipt of the certificate, recompile the application with the certificate and this will generate a new version of your application which can then be installed on the device for testing.

3.6.3 Development Environment

Setting up the environment for the Carbide C++ requires that Active Perl 5.6.1.631 is installed. If any other version except this exact one is used, then the developer will not be able to compile applications.

3.6.4 GSM cell tower access

When the initial concept for trilaterating position using GSM cell towers was conceived it was thought that it would be possible to scan for all cell towers in the area and trilaterate using the nearest 3-6 towers. However, this is not the case and during the development it was discovered that it is only possible to get the cell tower information of the currently connected cell tower. Further research has revealed that it would have been possible to do this if we were developing under the Windows CE platform but the Symbian S60 operating system has put in place restrictions that prevent a list of cell towers from being obtained.

3.6.5 WiFi MAC restriction

When we first developed our WiFi module we discovered that while we are able to get the SSID and the signal strength of the WiFi access point, we are unable to get the MAC address of it. This is because Symbian had also put in place a restriction that prevents access to some of the attributes. This problem has since been resolved and a patch was released that allows the developer to access this data, though rather crudely.

3.7 Future Considerations

ZigBee (IEEE 802.15.4)

ZigBee is the name of a new high level communication protocol using small, low power digital radios based on the IEEE 802.15.4 standard. It has being designed for devices that require a low data rate, a long battery life and is intended to be a cheaper alternative than other likewise technologies such as Bluetooth but not a direct competitor. It was decided that this is still an emerging technology and until there are enough real world devices that use this, we would not develop a module for it. However, although there is no current module for this wireless technology, due to the inherent open ended design of the ILC,, it would not require any major changes to the ILC as only an extra module would be needed to add this functionality. Such is the case with any future technology that becomes popular with mobile devices.

4 Conclusions

The main purpose of the ILC is to provide privacy sensitive, low cost and open source positioning to mobile devices. The initial use for this technology is to provide position estimates to a suite of e-Government applications developed as part of the ICiNG project. A prototype ILC has been developed that integrates GSM, WiFi, Bluetooth and GPS into two Symbian S60 applications that scan for wireless beacons in the area and based on a known database of beacons, provides an estimated or best guess location. In doing so, the ILC is designed in such a way that it attempts to deal with some of the issues relating to mobile positioning, in particular that of privacy and cost. Using the ILC does not incur any network costs in relation to everyday usage and access to the devices position is left to the discretion of the user.

The initial ILC prototype uses some simple trilateration algorithms to determine its position based on the currently visible beacons. The first prototype has been implemented but no test results are available at the time of publication. The next step in this research is to investigate some other algorithms with the aim of increasing the accuracy of the calculated location, the performance issues in terms of CPU load and battery life, using all or a combination of the technologies mentioned and by testing them in environments that have different types of wireless infrastructure, in particular the performance difference associated with using a sectorised WiFi infrastructure. Testing will confirm how each of the trilateration algorithms perform on mobile devices and if there are any performance issues, and if so, where do these issues arise and what modifications are needed to improve on them. Again, when we have concrete test results, we will publish them and further enhance the efficiency and usability of this location based system.

5 References

Bahl, P. and Padmanabhan, V.N., 2000. RADAR: An In-Building RF-based User Location and Tracking System, IEEE Infocom 2000, Tel-Aviv, Israel.

Ferris, B., Hahnel, D. and Fox, D., 2006. Gaussian Processes for Signal Strength-Based Location Estimation, Robotics: Science and Systems, 2006, Philadelphia, Pennsylvania.

Hightower, J. and Borriello, G., 2004. Particle filters for location estimation in ubiquitous computing: A Case Study. Springer Berlin / Heidelberg, 3205: 88-106.

Hightower, J., LaMarca, A. and Smith, I., 2006. Practical Lessons from Place Lab. IEEE Pervasive Computing, vol. 5, no. 3(Jul-Sept): 32-39.

Kilfeather, E., Carswell, J., Gardiner, K. and Rooney, S., 2007. Urban Location Based Services using Mobile Clients: The ICiNG Approach, GISRUK, Maynooth, Ireland.

LaMarca, A. et al., 2005. Place Lab: Device Positioning Using Radio Beacons in the Wild, Pervasive 2005, Munich, Germany.

Mokbel, M.F., Chow, C.-Y. and Aref, W.G., 2007. The New Casper: A Privacy-Aware Location-Based Database Server, International Conference on Data Engineering, Istanbul, Turkey.

Rooney, S., Gardiner, K. and Carswell, J., 2007. An Open Source Approach To Wireless Positioning Techniques, Mobile Mapping Technology, Padua, Italy.
Vossiek, M. et al., 2003. Wireless local positioning. Microwave Magazine, IEEE, 4(4): 77-86.