International Journal of Communication Networks and Information Security (IJCNIS) Vol. 1, No. 1, April 2009

A Framework for Preserving User Privacy and Ensuring QoS in Location Based Services using Non-irreversible Algorithm

Ch. Z. Patrikakis, M. N. Masikos, and A. S. Voulodimos

School of Electrical and Computer Engineering - National Technical University of Athens,
9 Heroon Polytechneiou Str, Zografou, 15773, Greece
{bpatr, mmasik, thanos} @telecom.ntua.gr

Abstract: In this paper, we address the issue of preserving user privacy in Location Based Services provisioning. This issue includes two controversial requirements, namely the protection of user's location privacy, and the Location Based Services QoS. In this context, we propose a framework that implements a non-irreversible algorithm, meaning that we cannot discover user's exact position based on provided services. The proposed algorithm has been implemented and integrated in a mobile services framework based on Microsoft's Virtual Earth platform.

Keywords: Mobile communications, Location Based Services, privacy, entropy**.**

1. Introduction

Mobile communications have shown a tremendous evolvement in the last decades. Currently, we are able to enjoy 3.5G services in numerous places around the globe. This boom was highly facilitated by the huge increase in the number of subscribers all over the world. According to the International Telecommunication Union (ITU) the estimated number of mobile subscribers increased from 16 million in 1994 to 1,758 million in 2004. Nowadays, is estimated that there are over 2,000 million mobile communications subscribers worldwide. However, this great market share in combination with the fact that mobile services are offered in adequate quality, render useless the need for further development. Only the introduction of innovative, demanding and appealing (to subscribers) services could change the scene.

Location Based Services (LBS) fulfill all the requirements for serving as the vehicle that will drive this change. LBS describe services that *"send custom advertising and other information to cell-phone subscribers based on their current location"*. This information about the user's location affects significantly the quality of the service offered. For example, a stranger in a city could be easily informed for nearby restaurants based on her location coordinates.

This type of services becomes more and more popular among the subscribers. Their spread is undoubtedly favoured by the significant progress that has been recorded in positioning techniques area. Several positioning techniques including user device based (e.g. GPS) or operator based (radiolocation or trilateration) provide accurate user positioning. Of course, outdoors the Global Positioning System (GPS) is currently the most accurate and precise positioning system. Additionally, several other techniques have been developed both for outdoors and indoors, based on several signal technologies, like RF, UWB, infrared and Bluetooth, and operate in parallel to GSM/UMTS networks, WLAN networks or sensor networks. Combinations of these positioning techniques can be used in order to improve accuracy and precision.

Location Based Services may have succeed in this way to give a boost to mobile communications, however new problems arose. User's position constitutes personal information and any disclosure and improper use of them violates user's privacy. This privacy violation includes not only spam advertisements (e.g. coming from a nearby shopping centre) but also the use of location info for prying on someone's personal life. Consequently, preserving user's privacy, especially as regards her whereabouts constitutes a serious task in location aware mobile computing applications. This task is contradictory to that of providing accurate location data and exact answers to user requests regarding mapping information and directions. Accuracy in data provided over LBS as answers to user queries can be considered as a measurement of quality of the offered services [1].

Therefore, there is a trade-off between user privacy and LBS quality. In this paper, we present a framework that tries to balance the contradictory forces of privacy protection and QoS provided. An extra requirement that is also fulfilled by the proposed framework is that of an open system, which can be easily deployed without the need of introducing new modules or specialized servers.

The rest of this paper is structured as follows: Section 2 identifies the related work and all the effort put into ensuring user privacy through different techniques. Section 3 describes in full detail the concept of the proposed system, while section 4 goes one step further in reporting on the first implementation attempt. Finally, conclusions and future work are presented to the reader in section 5.

2. Related Work

Regarding protection of user's location privacy, several methods have been proposed based on blurring or noise insertion [2]. An indicative example of such techniques is given in [3] by Kido et al. that propose the transmission of several fake (or "dummy") locations to the service provider, so that the latter returns an answer for each of the locations. However, the use of blurring or noise insertion techniques requires extra computational effort that leads to the request of additional (unnecessary data)

from the service provider, in order to hide the user's exact preference. In cases of mobile devices and wireless connectivity, this excessive request for practically useless information leads to higher costs, while introduces an extra overhead in database searches. Furthermore, in the case of consecutive questions, special care must be taken so that the common set of fake locations between the database queries is not null or small enough to reveal the location of the user.

Another technique is that of hiding user's location information by mixing it with the same information provided by other users in order to achieve a certain level of "anonymity" [4]. In [5] Gruteser and Grunwald, propose an algorithm that deploys region quad-tree cloaking, trying to achieve k-anonymity in terms of spatial and temporal terms. The algorithm uses a recursive subdivision of location data around the user, up to the point that the selected quadrant includes a number of users below k-min, and then uses the previous level as the cloaking region. This approach, however, does not take into account privacy issues regarding personal preferences (i.e. if a user asks for jazz clubs on a specific area, then it may be assumed that she is a jazz music fan), while no threshold related to quality of returned information exists. Furthermore, the region required to achieve an adequate level of anonymity may be quite large, especially if the k-anonymity parameter is set by the user. This is because the number of system users that have similar preferences in order to be included in a k-anonymous area may be scattered around a very large region. Finally, this method requires the deployment of an anonymizing server that collects information from many users, and acts as a proxy of the requests to the service provider introducing extra complexity and delays in possible implementations of systems based on this approach. In [6] Gruteser and Liu investigate disclosure-control algorithms that conceal user's positions in specific areas and hide path information that shows which areas they have visited.

In [7], the CliqueCloak cloaking algorithm is proposed by Gedik and Liu. The algorithm operates at a per-user basis, taking into account personal privacy settings and QoS requirements in terms of cloaking latency and cloaking region. All requests from users are anonymized through the use of an undirected graph consisting of user requests that have not been anonymized yet. Again, an anonymizing server is used that cloaks a request by combining a user's request with other, already existing, requests within the same area. So the existence of similar requests is a prerequisite for the operation of the cloaking algorithm. The method is effective for small numbers of users (around 5, which is considered a good level of anonymity), although the complexity of the algorithm is high, while in cases where anonymity cannot be achieved user requests are dropped.

In [1], Mokbel et al follow the k-anonymity model, trying to provide a framework that meets the demands for privacy and quality in database queries for location based services. Their model is based on the use of a location anonymizer and a privacy-aware query processor, for hiding information about a user's location. Although their approach provides a very good framework, over which hiding the actual location of a user can be established, it is still based on the periodic collection of information from other users, thus requiring continuous communication for location reporting, and on the existence of specific infrastructure (location anonymizer, privacy-aware query processor) making it vulnerable to attacks. Furthermore, regarding the issue of quality in the returned results the criteria selected for the appropriate level of abstraction in user location reporting are based only in k-anonymity and map detail levels, without taking into account the number of returned results. The latter is important, as a small or big number of returned results may compromise the effectiveness of the service.

Other techniques use abstraction layers in reporting information about the user's location. This abstraction is equivalent to the increase of entropy as regards positioning or representational accuracy [8]. The higher the level of entropy is the better protection of user's privacy is achieved. To address this, many proposals assume that a central entity (e.g. anonymizing server) is deployed. The exact user position is reported to this entity, while increase of entropy is provided through the augmentation of the location area, which includes the exact user location, by the anonymizing server. In cases where anonymity enforcement is based on aggregation of data from many users, the effectiveness of the methods is heavily depending on the existence of related information from other users in order to be successful (e.g. k-anonymity techniques), while the anonymity server constitutes a possible point of failure in cases of misuse or inadequate security. Furthermore, the assumption that the user location is calculated at the user's device (e.g. through the use of GPS technology), completely disregards the fact that Mobile Network Operators (MNOs) can estimate user's position via network-based positioning techniques such as trilateration or CGI-techniques (Cell Global Identity).

In our approach, we adopt the principles of Mokbel et al. [1] regarding the requirements for quality and privacy in LBS provisioning, but contrary to their work and to work based on k-anonymity ([9], [10], [11]), we rely on a method for hiding the location of a user purely on geographical data, without involving information reported by other users. Furthermore, we enhance QoS via taking into consideration the number of returned results within user defined thresholds.

3. The Proposed Framework

A very naive privacy ensuring approach as regards reporting of geographical location would be the following: the LBS server provides all available information for a large area to the terminal (e.g. all available drugstores in a large city) and local processes choose the best information for the user. As a large area, we identify a region large enough to guarantee that the location of the user is not identifiable. Such an approach can easily ensure privacy; however, a large quantity of useless data has to be transferred over the network wasting useful resources. The problem to be addressed here is the identification of an area large enough to ensure user position privacy, but small enough to reduce the transmission of unnecessary information. Furthermore, the number of returned results for the selected area should match user requirements as

regards minimum and maximum numbers, with minimum identifying the least of results that should be returned for a specific query, and maximum, the number of results above which information is useless.

In the following, we describe a framework for LBS provisioning that addresses all the above requirements. The framework consists of an enhanced LBS server and mobile devices equipped with GPS modules for location tracking, while no intermediary servers are needed, making the deployment of the framework easy and cost effective.

3.1 The map framework

Firstly, we must define a map framework in order to group and organize location dependent information. For this, any system that uses a hierarchical data structure to organize a geographical object space may be used, such as for example a region quadtree [12], [13]. However, in order to be able to present the proposed methodology based on a specific implementation framework, the Microsoft's Virtual Earth System (a.k.a. Windows Live Local) for providing the maps has been selected and will be used in the rest of the paper. This selection has been based on the inherent support for different levels of abstraction and accuracy regarding positioning.

Therefore, a brief presentation of the features of the Virtual Earth Tile System [14] that will be used is necessary. Virtual Earth, using the Mercator projection, splits the world map into square tiles with a size of 256x256 pixels. Currently, 23 levels of detail are supported. At level 1, the world map is 512x512 pixels and consists of 4 tiles (tiles 0, 1, 2 and 3). Each time the level of detail is increased by 1, the map width is doubled. In general, map width can be calculated as: map width = map height = 256 * 2 ^{level} pixels. For example, at level 2 the map width will be 1024 pixels, namely the map will consist of 16 tiles. Each level 1 tile is analyzed in 4 tiles at level 2 (tile 0 will be split into tiles 00, 01, 02 and 03), 16 tiles in level 3 etc. In this way we group information into these specific tiles.

These tiles have a specific ground resolution. This ground resolution indicates the distance on the ground that's represented by a single pixel in the map. For example, at a ground resolution of 10 meters/pixel, each pixel represents a ground distance of 10 meters. The ground resolution varies depending on the level of detail and the latitude of the tile. Using an earth radius of 6378137 meters, the ground resolution (in meters per pixel) is provided from the following formula:

$$GR = \frac{\cos(\frac{L \times pi}{180}) \times EC}{map\_width} \frac{meters}{pixel} \Rightarrow$$

$$GR = \frac{\cos(\frac{L \times pi}{180}) \times 2 \times pi \times 6378137}{256 \times 2^{level}} \frac{meters}{pixel} (1)$$

where

GR = Ground Resolution
L = Latitude
EC = Earth Circumference

Since each tile is 256x256 pixels the required tile should satisfy the following inequality:

$$256 \times GR \geq Diameter\_of\_circular\_range (2)$$

For example, if we assume that a subscriber is located at the equator and that the circular range has a diameter of 100 meters then inequality (2) should hold:

$$(2) \Rightarrow \frac{2 \times pi \times 6378137}{2^{level}} meters \geq 100 meters$$

$$\Rightarrow 400750{,}1669 \geq 2^{level} \Rightarrow level \leq 18.6 \approx 18$$

From the above, it is obvious that based on (1) and (2) we can easily calculate a first estimation for the abstraction layer, based on the selection of a minimum circular surface area which should be reported as the user's position.

The dendrogram is in fact a quadruple tree that represents the square tiles. Quadtree keys (or "quadkeys") are used in order to optimize indexing and storage of files. They represent the two-dimensional XY tile coordinates at a specific level of detail with a simple string. In that way a single tile at a particular level of detail is identified by a unique quadkey. The length of the quadkey denotes the level of detail. The quadkeys dendrogram corresponding to the Virtual Earth Tile System is depicted in Fig. 1. Note that for reasons of presentation, only the first 3 levels of the available 23 are depicted.

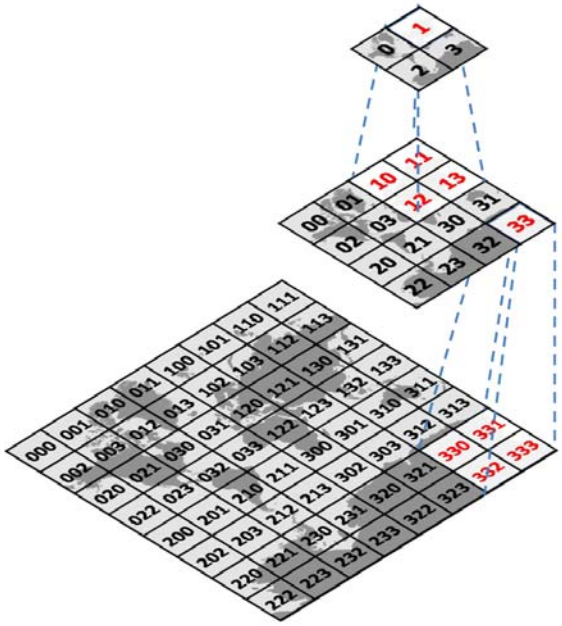

Figure 1. Quadkeys dendrogram for the Virtual Earth Tile System.

Coming to the issue of information representation, each partition level in the dendrogram corresponds not only to a different level of accuracy regarding user positioning but also to a different level of QoS. The concept of entropy can be used in order to describe accuracy [8]. Information entropy has been used in many fields, such as in ecology, in order to determine species' diversity [15], in sociology, to examine societal evolution, in taxonomy, to evaluate classification methodologies [16] and in robotics, to measure multi-agent system diversity [17].

In a similar way we characterize as entropy increase the transition to a higher level in the tree's hierarchy. Such a transition increases the ambiguity regarding user's

position and decreases the provided QoS. QoS, of course, refers to the degree that provided information matches user preferences and has nothing to do with the details of the provided information. For example, in case of low QoS the LBS server will return some drugstores that are located faraway from the requestor, which has nothing to do with the provided info concerning drugstores (address etc.).

3.2 The privacy ensuring mechanism

We propose a mechanism that supports the automatic selection of the best level of abstraction, namely the appropriate hierarchy level in the dendrogram that satisfies both the need for privacy as this has been expressed by the user and at the same time the need for quality in terms of a personalized, targeted set of returned results to the user's query. To provide this mechanism, we describe how the boundaries are defined by technology and problem definition itself.

Nowadays, several positioning solutions exist that can locate a mobile user with increased accuracy. These positioning technologies are classified in network-based and terminal-based ones depending on whether position is estimated in the network or in the terminal side. For example, CGI (Cell Global Identity) [18] is a network-based positioning technique, while GPS is a terminal-based positioning technique. On the other hand, there are techniques that are both network- and terminal-based, like Assisted-GPS [19]. Taking into account the network-based techniques we could support that the accuracy provided by these techniques determine a lower accuracy limit in the proposed algorithm. More precisely, since a user can be located in a certain area (the range of which depends on the accuracy of these techniques) there is no reason to report to the LBS server less accurate positioning data for the location of a user. In other words, there is an upper bound in the level of abstraction that the user may request as to ensure her position privacy. Consequently, this technological boundary corresponds to the lowest level of location details that the user may request to be sent to the system, since any attempt in trying to hide or blur further information about her position is pointless.

Having set a lower bound in positioning information provided, we should check if a corresponded upper bound exists. Indeed, this exists by default and it is defined by the level of detail that the Virtual Earth Tile System can provide (namely the 23 levels of quad keys). As a result the proposed mechanism cannot provide location dependent information for a smaller area than the lowest square (highest accuracy) tile of the 23rd level of the dendrogram depicted in Fig. 1. Tiles of this level will be referred as MinTiles in the rest of the paper.

Having defined the lower and upper bounds, we set an area of the dendrogram in which user location reporting can take place. According to the proposed mechanism, the user defines a geographical bound that ensures privacy according to her preferences. The GPS module attached to user's device provides an accurate position estimate. Since commercial devices advertise accuracy up to even 1m, while practical tests indicate that in practice accuracy of devices is much less, we have tried to make an assumption on average accuracy that is as close to real life situations. In this, the results of Wing et al in [20], in which they tested the accuracy and reliability of consumer-grade GPS receivers in a variety of landscape settings, have been adopted. According to these tests, the best accuracy, achieved in open sky conditions and in non urban areas was measured at an average of 5m. However, this estimation is blurred in order not to reveal exact user's position to the LBS server. The user sets the minimum area including her position that she wishes to be announced to the LBS server. Of course, there is no point in setting as a bound an area larger than the one estimated from network-based positioning technologies. Consequently, the following mathematical relation must be taken into account:

$$Area\_Limit\_Set\_by\_User \leq Network\_Pos\_Tech \quad (3)$$

After ensuring user privacy based on her preferences, we must examine how QoS is affected. QoS refers to: (a) the LBS returns the best answer to user's query, for example the closest drugstore, and (b) the LBS returns at least one answer (or the minimum number of answers that are acceptable according to user preferences) to user's query. Regarding the first point, the LBS server may not return the best answer as this requires the disclosure of user's exact position. To address this issue, a processing of all returned results from a user query, should be performed on the user's device, so that sorting of results according to the exact user's position is performed. This type of processing can be performed in the user's device without compromising user's location privacy. Regarding the second point, it is a fact that the user defined area for a specific query, might return no results. For that reason, the user must define a minimum number query results that the system has to return. If the LBS server does not find the minimum number of points of service in the determined area, then it will continue searching in a larger one. This tactic ensures that the user always get an answer from the LBS server. Then it depends on her, whether she will choose any from the returned points of service.

Based on the observations made previously, the proposed algorithm consists of the following steps:

i.The user sets the minimum detection area that will be revealed to the LBS server and the minimum and maximum number of returned points of interest.

ii.Based on minimum detection area and on formulas (1) and (2) the abstraction layer is determined. Afterwards, user's position is determined via the GPS module and the terminal software searches for the tile (of the previously determined abstraction layer) that includes the user's position.

iii.The quadkey corresponding to the previously determined tile together with the minimum and maximum number of points of interest required by the user are transmitted to the LBS server.

iv.The LBS server searches for points of interest in the quoted tile. It evaluates the number of returned results, and if this is smaller than the desired number of results, it advances to the higher abstraction level (details on this advancement procedure follows in Section III.C) that contains the tile where user is located. This process is

performed until the number of returned results is within the desired limits.

v.The results are returned to the user. In case of a large number of returned results the software running on the terminal may sort them and propose to the user the best ones depending on her position.

3.3 Increasing the search area to satisfy user QoS requirements

According to the proposed mechanism the LBS server should run a process of identifying the smallest area that satisfies the previously stated requirements (we name it "Best Fit Area – BFA"). Since the exact position of the user is not known, the area that the user's device has reported as the user's position should be the starting point. From this, the LBS server starts an iterative process of identifying the BFA. An easy but naive approach would be to start increasing the level of abstraction in user positioning by climbing up the quadkeys dendrogram, and checking on the number of results reported. This approach, though it is simple and easy to implement, does not return the best results, as there is the possibility that as we move up the quadkey tree, the center of each tile is shifted away from the actual user position. Therefore, the results reported do not include points near the actual user's position, as these points belong to tiles much higher in the quadtree hierarchy, and the iterative process of increasing the search area has to pass through several steps before these are included. The example depicted in Fig. 2 gives an idea of the problem.

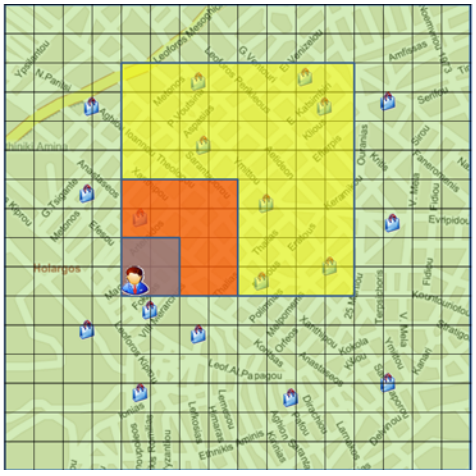

Figure 2. Possible problems when increase of the search area is based solely on the tiles hierarchy.

Let us consider that a user is located somewhere inside the inner purple rectangle. The user asks for the location of nearby shopping centers. Let us assume that according to her privacy settings the purple rectangle is reported to the LBS as her position, and also that the minimum number of results expected is higher than the points included in this rectangle. The LBS, trying to satisfy the user's needs, increases the area on which the user's query is applied, following the next level of abstraction according to the quadkeys dendrogram, resulting in the search areas of Fig. 2 depicted by the orange and green rectangles. We see that in each step the surface area is increased by four times compared to the previous one. In our example, the final area is 16 times larger than the initial one, while the user is positioned in one of the corners. Therefore, the results returned are not calculated around the actual position of the user.

Due to this problem, we follow a different approach for defining the BFA: Starting from the initial tile reported by the user's device as the one holding the user's position, the LBS server increases the area where the query is applied, by a shell build around the last calculated area. The shell is constructed by using MinTiles, placed around the last calculated area. If the search area is a square consisting of n^2 MinTiles, the next one is $(n+2)^2$ MinTiles large, meaning that the search area is increased by $4(n+1)$ MinTiles larger. Following this approach, the search area is enlarged in each step, around a central block in which the user is located. Fig. 3 depicts this process. Again, the blue, red and green areas indicate the different search areas.

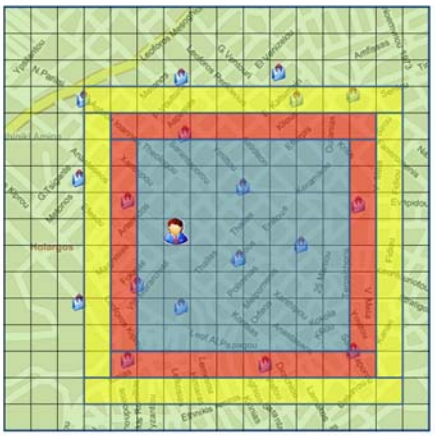

Figure 3. Selecting the appropriate level of map details according to quality and privacy.

3.4 Form query's response

Finally, after defining the search area for points-of-interest the LBS server has to search its database for points-of-interest that satisfy user's query. Points-of –interest selection is not based on the calculation of distances between the user's position and points-of-interest contained in the database. Instead, each point is characterized by a value corresponding to the MinTile in which it is located. Therefore, screening of these values against the values of tiles that cover the search area is used by database searching algorithms. To improve the search processing time, a check for tiles that can be grouped into one of higher layer is performed before performing a search process (e.g. if the search area includes small tiles of n level that can be grouped into one tile of higher level, this is performed before performing the search process to the LBS server database).

Finally, the points-of-interest returned to the user's device can be further filtered and presented in order of distance, by calculating the actual distance of each point from the exact location of the user. Since this is performed in the user's device, accuracy of reported results is high, while no information about the actual user's position is reported to the service provider, thus protecting user's privacy.

4. A Practical Implementation

The proposed methodology for providing location based, privacy preserving, mobile services, has been materialized into a framework deployed over mobile devices, and has been put under test in real life conditions. The architecture is depicted in Fig. 4.

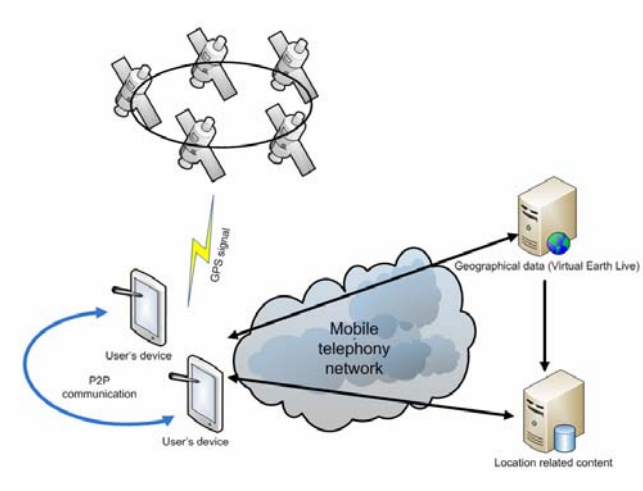

Figure 4. Architectural design of the implemented framework.

A mobile computing platform for offering personalized, location based services named PLASMA (Personalized, Location Aware Services over Mobile Architectures) [21] has been designed and implemented, deploying the described framework.

The platform is based on the use of PocketPC mobile devices with mobile telephone capabilities, and is using Microsoft's Virtual Earth Live system for mapping and location tracking purposes.

The client application is based on the deployment of four different APIs, as they are depicted in Fig. 5:

- A sensor API that is used for getting positioning information based on input from GPS and RFID reader modules.
- A Service API that is used to handle interaction with the LBS server and Virtual Earth Live System in order to request mapping information and deploy queries.
- The Info display API that is used to display mapping information and visualize the query results.
- The P2P communication API that is used to support peer to peer communication between devices for location information and direct exchange. This is used to support direct communication of private information between trusted devices, without the mediation of the server.

The platform supports the personalized provision of context aware services In order to be able to form and deploy queries, the mobile device application, using the service API, can identify all available information available to the LBS server. Once information about a specific point of interest needs to be transformed into a query, personalized user settings are used in order to determine the level of ambiguity the user desires to be used in the query. Once this has been determined, then the query, incorporating personal information is addressed to the LBS server, while hiding the exact user's position.

An example of this is the case where the user wants to find out about possible points for eating, near his present location. Her personal settings stored in the mobile device identify her as a vegetarian, while her privacy settings indicate that her position is not be revealed to the system at an accuracy level below $200m^2$. Based on this information, the application running on the user' s mobile device addresses a query about vegetarian restaurants in the corresponding tile of the Microsoft Virtual Earth Live system that is bigger than $200m^2$ and in which the user is located. The query to the LBS server is accompanied by a minimum and maximum number of results that the user wishes to receive and the information fields that the user would like to receive for any point that matches her query. The server runs the query, and checks on the returned results. If this number is less than the minimum indicated by the user, then the method for increasing the search area in which the query is applied is used, until a suitable area for answering the query is found. Once the number of returned results is inside the limits identified by the user, they are returned to the mobile application. There, they are filtered against user preferences (these are private settings used only in the mobile device) in order to personalize the list of results, e.g. exclude restaurants that have been identified as very expensive. Finally, the returned points-of-interest are sorted according to distance, which can be easily calculated in the user's device, as the exact location of each result together with the user's position are known.

As regards the storage of the information points in the system, each point contains the following mandatory fields: latitude, longitude and quadkey value. The first two comprise the actual location point in the map, while the last is the identification number of the MinTile in which the information point is located. This assists in serving the queries based on quadkey matching between the user's position and points-of-interest. This matching is performed only in that part of the quadkey that corresponds to the level of detail of the user's location reporting tile.

Finally, hiding the user's position can be applied also in P2P location information exchange. Using this feature, a user can directly report her position to other users either using latitude and longitude, or the corresponding to the user's privacy level tile that includes her position.

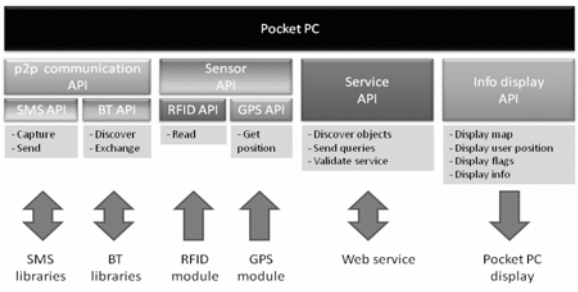

Figure 5. Architectural design of client application.

5. Conclusions and Future Work

In this paper, we presented a framework for offering location based services while preserving user's privacy and ensuring QoS. The proposed methodology differs from the existing approaches in several aspects. Firstly, the processing takes place at the user's device and no anonymity intermediary servers are needed. Secondly, the capability of hiding the user's location is unbound from the existence of location information about other users and is based on spatial nature techniques that do not require any processing based on time. As a result, the implementation of the method is easier and it can be used complementary to other approaches (e.g., in conjunction with any k-anonymity approach).

Protection of private information is guaranteed, since the exact location of the user can be found only in her device. Also, the reduction of spatial information accuracy is performed by algorithms deployed in the user's device, and completely independent of time information, thus unbinding the level of privacy achieved from information about other users available at the time. Finally, the lack of an intermediary server offers an open framework, where location based services can be offered and combined with third party data services, completely free of integration issues or incompatibility problems.

What's more, although the proposed method has been inspired by the need of location privacy protection, it can be extended to all aspects of user personal information protection. Via modeling user profile preferences into a hierarchical data structure by using data clustering techniques, the same method can be used for hiding the exact user profile, and report user preferences by using different levels of abstraction, that do not reveal specific user characteristics. For example in case of film preferences two users that like western movies and martial art movies respectively are both reported as "action movie" lovers, which could also include war movies, sci-fi movies etc. Currently, the authors are working towards the provision of a unified platform (based on the framework presented above) that will be able to offer a truly personalized experience (location and context services) without compromising personal information privacy.

Trying to classify the types of queries that are used for supporting location based services, Mokbel et al. in [1] have identified three distinct types: (1) private queries over public data, eg., "where is the nearest gas station" where a private entity, such as a person, requests information that is publicly available, such as the addresses of gas stations, (2) public queries over private data, e.g., "what is the number of cars in a certain area", where a public entity, such as the highway patrol, asks for private information in terms of location, and (3) private queries over private data, e.g., " where is my daughter now", where a person requires to find out private information about another person or persons.

The above classification is based on the assumption that queries are location oriented (this is the reason why no public queries for public data are identified, since the idea of location awareness regarding the status of individuals is missing from this type). However, user queries usually include more private data than their location, e.g., "where is the closest gas station of my favourite gas company that accepts a certain credit card?", or "which is the closest restaurant for vegetarians that has a playground for kids?". In this generalized type of queries, the parameter of personalization and private data is introduced making the privacy problem more complex. Up to now, all approaches on the issue of privacy in location based services ignore this parameter. The proposed methodology can be easily extended to incorporate privacy of personal data as well as location (in fact the authors are currently implementing a platform that extends the ideas presented in this paper, so as to include personal preferences based on data clustering techniques)[22].

References

[1] M. F. Mokbel, C. Chow, and W. G. Aref. "The New Casper: Query Processing for Location Services without Compromising Privacy," In *Proceedings of the 32nd Conference on Very Large Data Bases,* 2006.

[2] M. Duckham and L. Kulik, "A formal model of obfuscation and negotiation for location privacy," In *Pervasive*, pp.152-170, 2005.

[3] H. Kido, Y. Yanagisawa, and T. Satoh., "Protection of Location Privacy using Dummies for Location-based Services," In *Proceedings of the IEEE International Conference on Pervasive Services*, 2005.

[4] L. Sweeney. "Achieving k-anonymity Privacy Protection using Generalization and Suppression," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5), pp. 571–588, 2002.

[5] M. Gruteser and D. Grunwald. "Anonymous usage of location-based Services through spatial and temporal cloaking," In *Proceedings of the First International Conference on Mobile Systems, Applications, and Services*, 2003.

[6] M. Gruteser and X. Liu. "Protecting privacy in continuous location - tracking applications," *IEEE Security & Privacy*, 2(2): pp. 28-34, 2004.

[7] B. Gedik and L. Liu. "Location Privacy in Mobile Systems: A Personalized Anonymization Model," In *Proceedings of the International Conference on Distributed Computing Systems*, 2005.

[8] X. Jiang and J. A. Landay. "Modeling privacy control in context-aware systems," *IEEE Pervasive Computing*, 1(3), pp.59-63,. 2002.

[9] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. "Incognito: Efficient Full-Domain K-Anonymity," *ACM SIGMOD International Conference on Management of Data*, 2005.

[10] R. J. Bayardo and R. Agrawal. "Data Privacy through Optimal k-Anonymization," In *Proceedings of the IEEE International Conference on Data Engineering*, 2005.

[11] L. Sweeney, "k-anonymity: A Model for Protecting Privacy," *International Journal on Uncertainty,*

Fuzziness and Knowledge-based Systems, 10(5), pp. 557–570, 2002.

[12] A. Klinger. *Patterns and Search Statistics, in Optimizing Methods in Statistics*, pp. 303-337, J.S. Rustagi, FEd., Academic Press, New York, 1971.

[13] H. Samet, "The quadtree and related hierarchical data structures," *ACM Computing Surveys*, 16, pp. 187-260, 1984.

[14] Virtual Earth Tile System webpage, URL http://msdn2.microsoft.com/en-us/library/bb259689.aspx, retrieved February 4th, 2008.

[15] D. Lurie, J. Valls and J. Wagensberg. "Thermodynamic approach to biomass distributions in ecological systems," *Bulletin of Mathematical Biology*, 45(5), pp.2869-872, 1983.

[16] N. Jardine and R. Sibson. "Mathematical Taxonomy," John Wiley and sons, 1971.

[17] T. Balch. "Hierarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity" in *Autonomous Robots,* 8(3), pp 209-238, Springer Netherlands, 2000.

[18] E. Trevisani and A. Vitaletti. "Cell-ID location technique, limits and benefits: an experimental study," In *Proceedings of the Sixth IEEE Workshop on Mobile Computing Systems and Applications*, pp. 51-60, Dec. 2004.

[19] M. Djuknic and E. Richton. "Geolocation and assisted GPS," *Computer*, 34(2), pp. 123-125, Feb 2001.

[20] M. Wing, A. Eklund, and L. Kellogg. "Consumer-Grade Global Positioning System (GPS) Accuracy and Reliability", *Journal of Forestry*, 103(4), pp. 169-173, 2005.

[21] The PLASMA (Personalized Location Aware Services over Mobile Architectures) project web site, National Technical University of Athens, URL: http://www.telecom.ntua.gr/~bpatr/staticcontent/PLASMA.php , Retrieved January 27, 2008.

[22] P. N. Karamolegkos, Ch. Z. Patrikakis, N. D. Doulamis, E. Tragos, "User – profile based communities assessment using clustering methods," in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications* , 2007.