Presetting with QoS Architecture to Enhance LBS

Miada A. Almasre
Department of Computer Science
College of Computing and Information Technology
King Abdul-Aziz University
Jeddah, KSA
malmasre@kau.edu.sa

Jonathan Cazalas
Department of Computer Science
College of Computing and Information Technology
King Abdul-Aziz University
Jeddah, KSA
jcazalas@kau.edu.sa

Abstract— **Location-Based Services (LBS) send and receive real time information, which users access via their mobile phones. Enhancing the Quality of Service (QoS) of mobile location based services is a vital research area that requires attention in order to enhance such applications. One such method of improvement is by using a pre-setting process, which simplifies the restoration of data and guarantees an immediate response to user requirements as well as needing less calculations. There are some important requirements (presetting with select valid region, strong data indexing with less storage BQR*-tree, more storage with local servers) that will support roaming mobile users by utilizing more storage and less computations. The general aim of this paper is to propose a distributed database architecture with an algorithm based on a presetting process that supports lesser communication costs, while maintaining the quality of service. Additionally, the paper will present an overview about using WiMAX instead of WLAN in moving object and spatial or non-spatial queries.**

***Keywords-component; Non-Spatial range* queries, *Moving Objects*, *Location Base Services*, *Mobile Computing*, R-tree *structures*, *BQR tree*.**

I. INTRODUCTION

Location-based services (LBS) emerged as an essential group of communication services, which provide content and services based on the user's physical location as well as by non-spatial information [1]. Moreover, LBSs are utilities that utilize their own competency to create an influx in location awareness and to allow for easier user interactions. This furthermore, permits them to acclimate to a particular environment. These services are reachable via mobile devices through networks, while employing the aptitude to use the actual position of the mobile device. Similarly, Location-Based Mobile Services (LBMS) have the same ability where users may access the information provided from such services using their mobile devices [2].

A number of LBSs usually depend on the performance of assessing continuous range or k nearest neighbor (KNN) queries. These queries consistently regain data regarding the moving objects currently positioned within a spatial query region of interest, identified by the client [3,4]. As expected, this functionality needs mobile devices with high performance specifications or a central trusted server whose compute capability can handle the potentially tens of thousands of continuous queries. And in fact, with over six billion mobile users in 2014, many of whom use devices with dual-core, quad-core, and now Octa-core processors, it does seem that mobile devices will indeed dominate the future [5].

In spite of the developments, both in the fields of mobile computing and LBS applications, there are still issues that need to be addressed in regard to Quality of Service (QoS). Points of interest include, amongst others, the following issues: slow or limited bandwidth, limitations with device storage, and computation and processing requirements clearly beyond the capability of the device. What needs to be done then is to enhance Location-dependent query processing to assure the QoS. This is required in cases where the sufficiency of data is decided by the QoS. If the service coverage, bandwidth, or storage is low or limited, then retrieved data will not be of use to the users. For example, a mobile user tries to access location-based data about specific landmarks in a particular region and is hindered by bandwidth, which does not support the retrieval of exact locations of these landmarks. The QoS, in this context, affected output. What if the user has the option of caching or pre-loading data (information or maps) prior to going to the specified location? Or what if the users has option of easily accessing local servers' storage, which provide data about objects near him?

In such a situation, the user will use less power consumption, because the mobile device will not submit as many queries as expected to the LBS server. This will also decrease the heavy load on the server, because there is no need for a continuous query, one in which is run continually on the location-based server.

For the purpose of our research, we recommend a novel presetting method, wherein a user will send to the server the data requirements before visiting a potential query region. The result is a reduction in overall communication costs, as said user can simply perform queries on the cached data, which was received prior to leaving for the trip. Luckily, within the domain of our reality, the queries of a continuous query for the most part, demonstrate spatial locality [6, 7]. Therefore, not only caching the result of the query was proposed, but previous studies also discussed caching the analogous Valid Region (VR) based on the client side. Additionally, VR computation algorithms [8, 9] only requires a solitary index search. Yet more computation redundancy could happen where the VR of one object can be utilized to answer all the queries who question that object [10].

Therefore, we see the need to develop an effective strategy to reduce the valid region computation, or eliminate it completely, as said strategy will surely enhance many mobile applications.

In an ideal situation, this would guarantee the QoS even if the user encounters some other limitations. A person in charge of a practical field mission could need such applications with high QoS. For example, an urban planner involved in building a large regional structure (such as an amusement park, a sport city, etc.) needs to communicate with some visual objects in real time during the execution of his mission.

Continuous query processing in location based services is important and distinguished from traditional data streams [11]. Continuous queries over spatio-temporal data streams need more management, because queries, as well as spatial data, hold the potential to unceasingly altercate and have a series of data updates, as opposed to traditional data streams. Similar to customary data streaming applications, memory is always a limited source. That said, continuous, spatio-temporal data streams bring with them additional challenges, with communication costs and network bandwidth at the forefront [12,13].

Recent developments in wireless technology have positively impacted high quality of service. With the advent of wireless, wide area network connections (WIMAX) for example, such applications set high quality standards. Hence, we can say that providing high quality LBS applications could be achieved via presets and the use of a WIMAX.

All of that this can be considered in the context of spatial data structures, as they utilize strong indexing schemas, thereby requiring less communication or computation. In most of the previous studies, researchers used the R-tree and its variants as a basic indexing of a regions information. Other works made changes in the node value, such as R*-tree [14], which works for the various diverse kinds of queries and operations. Additionally, researchers have proposed the Bit-vector Query Region tree (BQR-tree), a method for indexing queries based on query regions, and it augments each node with non-spatial information in the form of a bit-vector.

The concept of resident domain has been studied extensively. This has been done along with extensions that include a novel query indexing structure, Query Region tree (QR-tree). This notion indexes queries centered on the original query regions rather than the monitoring regions and allows the server to allocate each individual moving object a more significant resident domain along with the original query region. [15] In addition, with non-spatial selections, they present a Bit-vector Query Region tree (BQR-tree), a further edition from the QR-tree. BQR-tree augments the QR-tree by storing non-spatial information inside a bit-vector for each node [15].

Many useful LBSs, including mobile advertising, usually depend on the functionality of estimating a continuous range query (CRQ). However, CRQs often send coordinate-updates to the server [16], so researchers [15] investigate a set of methods for scalable evaluation of conventional CRQs; they reported previous approaches such as Safe Region [17,18], Monitoring Query Management [19], and the Space Partitioning Query Index [20]. All these approaches consider spatial attributes only, and they assigned a much larger resident domain. Therefore, they cannot effectively deal with CRQs with selections on non-spatial attributes.

Skyline query processing is also proposed for LBS. This theory keeps into consideration both spatial and non-spatial characteristics of the objects being inquired about. This solution has been recently usein in use in recent times. Although it ut be pointed out that this has been accomplished with miniscule applications due to varying privacy trepidations along with inadequate exactitude of localization devices, though it received increasing attention [21]. The new query indexing structure that is proposed, the BQR-tree, is of practical benefit, as it deals with non-spatial selections and reduces the frequency of server contact by way of calculating a safe region [15].

While the BQR-tree was novel and enhanced the research in this domain, it is used a mapping function between object bit-vector and query bit-vector, thereby requiring the use of additional memory for both the non-spatial attributes and binary values. Therefore, we propose the Bit-Vector Query*- tree (BQR*-tree), which is created by augmenting the (BQR-tree) with additional information inside each leaf query node. Moreover, by using a local server instead of a centralized single server, which serves as single point of failure, this will increase the chance to achieve the high QoS focused on in this paper.

Towards this end, researchers deployed a proxy server approach and proposed to exploit the proxy server, which exists in company networks, to help in distributing the database among multiple locations. This additional device amid mobile clients and the LBS server [22, 23] helps by reducing the computation process on both, the main server and the moving object. Regardless of the success of proxy deployment and EVR provisioning, these come with various disadvantages such as sluggish progression in the valid region estimated list. [22] VR is another calculation. which consumes enormous battery power, as it uses continuous queries to the main server. [24, 25] This causes the orthodox structure to undergo various issues including challenges in wireless medium and heavy query load.

Due to the high cost of communication between moving objects and servers, we need a technique that reduces this communication load. One essential point resulting from many reports being sent by moving objects is their immediate current location. This, in turn, helps the server in continuously updating the results of queries. This process results in a major degrading of the system's performance. Some researchers proposed approximate range search (ARS) query and approximate continuous range search (ACRS) query; both of these queries use an approximate technique to retrieve estimated rather than exact results, which requires less computation and communication time query [26]. Others proposed Monitoring Query Management (MQM), which is always updated when the query is the same. It lays in two regions since the MQM depends on the region rather than the query [21].

We have to consider the wireless topology and its limitations (Wi-Fi converge a 300 ft. effective range from access point) [27]. Thus, as with all long-range communications, the WLAN is limited to providing a high quality of services in LBS. Therefore, using other technologies such as WiMAX will provide more coverage area with high QoS.

The contribution of our research paper can be abridged as follows:

1. We recommend an enhanced architecture with a presetting process to help cache LBS according to user presetting and consequently reduce valid region calculations.
2. We develop a presetting based algorithm by extending BQR-Tree (bit vector with QoS = BQR*-Tree). Specifically, we propose a multi-server approach, as opposed to a single central server, and we propose to extend the bit vector by using extra bits from the QoS classes (caching extra information in each node).
3. We statistically demonstrate that using the WiMAX technology enhances QoS through inputting IEEE-QoS-classes in mobile applications.

The rest of the paper is organized in the following manner. Section II reviews related work on spatio-temporal data from different research perspectives. We discuss some applicable, domain specific applications in section III. For the purpose of making this work self-contained, section IV briefly discusses the proposed architecture and its remunerations. Lastly, there is further discussion in V, followed by the conclusion and suggestions for future work in Section VI.

II. RELATED WORK

A number of researchers addressed the topic and proposed solutions, which vary according to the techniques being used. For example, some researchers implemented a number of time parameterized queries [28], retrieve the nearest object [29], V_*Diagram [30], which computes VRs by using the query location, the data in the nodes themselves, and knowledge of the search space.

Unfortunately, situations may arise wherein the LBS cannot provide certain features, such as calculating the user VR. Towards this end, Huang proposed a novel approach to window queries and continuous kNN queries by using proxies [31]. VRs are estimated by the proxies by exploiting both the spatial locality and the temporal location of the query.

Researchers from MG University [32] came up with an idea to improve system performance by studying the location aware proxies, caching, and using different query models. They proposed a framework with a hierarchical service database, which minimizes network cost, resulting in better QoS for the location-based services. In their tiered database structure with accumulated proxy, they simplify location-based service delivery for location dependent data to the users. The researchers have revealed a three-level tree structure of the proposed hierarchical service database, and they claim that this could be prolonged to various levels, based on the data management strategy. However, without experimentation, they could not prove that the extended three-level tree structure could improve the performance. We assume that it probably reduces search performance as the multiple levels might hinder results. Therefore, we proposed a two-level hierarchical service database in our research.

ZYPAD software was developed by Mobile3D and an Italian company. It is a tourist guide which uses advanced multimedia technology and was built to be used on PDAs and mobile computers. The software platform is organized in two main components; a desktop content management system and a specific client for ZYPAD wearable computers.

The guides were built in the following three steps; the first and second steps used the desktop application to create the contents then copied the tours and the mobile guide client to a miniSD memory card. By inserting the memory card into a wearable computer, the multimedia guide will be immediately ready for use. However, because this type of application requires specific hardware and advanced skills, it is not usable by everyone [33].

Another application that supports the presetting concept [34] is an application having two interfaces. The first one is the mobile interface that is designed with many built-in classes to enable the GPS to work correctly, alongside a wizard that asks if the user wants a specific region with specific tasks or information. The Second one is the admin interface (website) via which users update all location data. The user must select the region needed, then two main questions and the main menu of the system would appear. In the downloading step, the user can select some options to see information about many locations. Thus, for every location there are text and relevant pictures as archived while the real time information is displayed to the user upon request from the location server. If the user needs to go to, or know the direction to a location, distance or if they require any real time data regarding a definite service or the overall location of such service provider, the user will issue a range query. Through the admin interface, the website that manages some archived information and real time data on the data base server, the user can see all locations and work on any one of the main functions including but not limited to, changing the region of a map, any data, adding text, or images, movies...etc.

A. Data Structure

One of the most important ideas in the related work is the topic of data structure. The current algorithms that exist for continuous spatiotemporal query processing focuses primarily on data in disk-based index structures [35] such as hash tables [36], grid files [37], and the R-tree [38]. Otherwise memory-based data structures [39] normally work with realistic data sizes that can easily be fitted within the memory allocated for this purpose.

In most previous studies, R-trees and its variants are used as a basis of indexing of a region's information. Minor changes were introduced in the node values, such as R*-tree [14] holds for varying categories of queries and operations. Moreover, in [15, 40] they used the (BQR-tree), which indexes queries based on query regions, and it augments each node with non-spatial information via a bit-vector appended to each node.

The researchers [15, 40] classified the overlapping relationship between a spatial query region and a sub domain into four particular scenarios: covers, is enclosed by, partially intersects, and equals to. They represent non-spatial attribute values of moving objects and non-spatial values of queries as object bit-vectors and query bit-vectors, respectively. The BQR-tree is constructed using a recursive split of the whole workspace around the split threshold, where every sub domain is limited in its query regions based on their coverage, or partial intersection, with the sub domain. The threshold value is the minutest ability of all the objects in motion that are enumerated at the server.

The method based on bit vector index is suitable for low cardinality columns and can offer significant saving in terms of space [15]. Therefore, we use it with enhancements to store more details, by extending the BQR-tree to become a Sever Region Tree (BQR*-tree) that indexes the sever information in a specific region with the QoS that the server could provide.

A leaf node of the BQR-tree stores at most t entries of the form (qid, q.bv), where qid refers to a query q in the query table and q.bv is the query bit-vector of q. As well as leafs of QBR*-tree has the same information. Similarly, the non-leaf node in BQR-tree saves two entries of the form (ptr, N, N.bv):

- ptr refers to a pointer to a child node.
- N is a sub domain of the child node pointed to by ptr.
- N.bv is the node bit-vector of a node N.

While the non-leaf node in BQR*-tree stores extra information about the QoS that the user needs in any given region, the local server is used to predict the bandwidth that this region needs, allowing to facilitate the wireless communication management.

Wireless network supports the user to organize a computer network barring the need to route cable through the entire facility. With mobile applications, we have to be concerned about the wireless topology and the limitations (Wi-Fi converge a 300 ft. effective range from access point) [27]. Therefore, using other technology, such as WiMAX, allows for extended coverage areas.

WiMAX Forum developed a QoS architecture framework model, which backs both static and dynamic apparatus of the service flows. Moreover, it defines the various QoS-related functional entities in the WiMAX network and the apparatuses for handling the varying service flows and the policies that are related to them. In addition, this architecture backs the dynamic manufacture, admittance, initiation, alteration, and erasure of service flows [41]. The IEEE802.16 WiMAX standard offers categories for the prioritization of traffic and the following table shows the five QoS classes with a specific application [27]. Finally, the following table shows the five classes of QoS that.

TABLE I. WIMAX QOS SERVICE CLASSES

QoS Class	Applications
Unsolicited Grant Scheme (UGS)	Multiplayer Interactive Gaming, Studying or Training
Extended Real Time Polling Service (ertPS)	VoIP / Video Conference
Real Time Polling Service(rtPS)	Streaming Media
Non Real Time Polling Service(nrtPS)	Web Browsing and IM, Messaging Media Content Downloads
Best Effort Service (BE)	

The application column in the table represents an example of a class service that a user could requests.

III. SOME APPLICATION SCENARIOS

Extending LBS applications with high QoS is on demand in any competing institution that wants to provide high quality services. Along with the expansion of expertise in wireless and mobile computing fields, it has become necessary for both educational and governmental institutes to think about expanding their institution's activity outside the physical building.

GPS-based navigation systems have gained in popularity, owing to the fact that they allow people to swiftly discover new and unfamiliar areas. I sketched some possible scenarios for using LBSs that uses the architecture proposed in this paper.

- Outdoor task: a student with filed tasks can use such an application. The presetting technique will allow them to get the task and the location-based reminders or alerts (by the trainer, faculty or coach) while being onsite.
- Child kinetics outdoor games: using reminder or advertisement at sites, the child stores some instructions or reviews by generating a bundle containing all the information related to the game at a specific point. They can even review items, such as the text, ratings, and voice records etc.

The LBS fits in such applications, where after logging in, the user will be able to observe the main screen on which they will be given numerous alternatives. This works for both archived information and real time information. We should also point out that the archived information contains data regarding various site termini along with imageries. This information could be notes about a specific tasks, comprising of all the plans about achieving this task, guidelines, rules or even policies can be accessed by the users.

Therefore, the user can for archived data where the system will utilize web services to upload the information from DB on to the main server during this request. Once the user logs in, the system allows the data to be conveniently available to all users anytime and everywhere, reducing the need to continue server request during poor network coverage. The user can also request real time data using cache and local server. The system will also display information as graphical, text, and in media format.

IV. PROPOSED SOLUTION

A mobile user communicates with the local server database of his region. This leads to the routing of the query to the suitable level of the database structure. This is primarily based on the tye of query and the position of the target object. The proposed architecture shows the model of the real-time data and the archived information as a distributed system which contains several components working together. "Fig. 1" below provides architecture "level0" of the proposed solution.

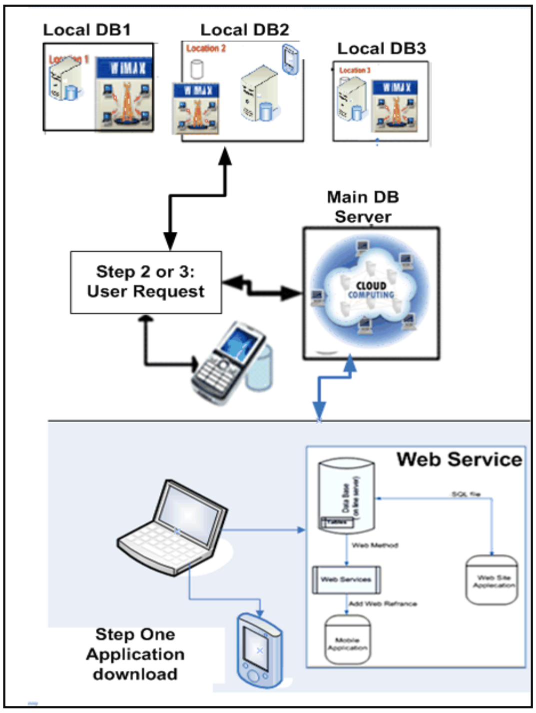

Figure 1. Level 0 of proposed architecture

The main DB server is the main part of the system that users interact with using either their personal mobile devices or local device. The user's interface is used to update information or services that are requested. The local server in each location has the unique application service to quickly answer all user requests and provide the QoS that the user needs. The WiMAX and base stations are the wireless backbone infrastructure of the system. The following flow charts, "Fig. 2" and "Fig. 3", respectively, provide more details about the general pre-setting procedures.

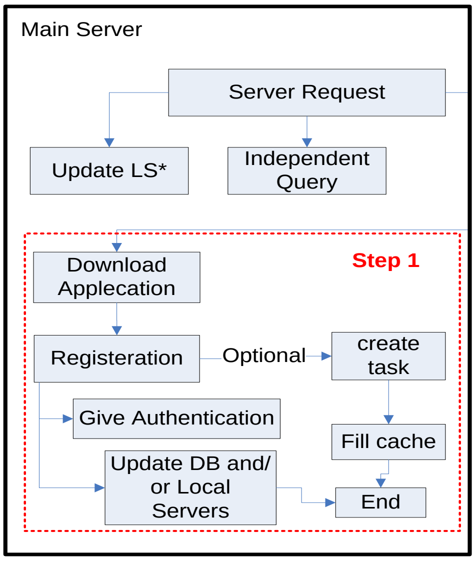

Figure 2. Step1 of flowshart

In this section, we are going to clarify the flow of the presetting process, which could be outlined as follows:

Step one: User requests main server to download the application. Then the user registers an account to receive authentication. If the user wants, he can start "creating" tasks, which fill the cache on his device. This step ends with the main server updating its database and/or the local server.

Step two: When the trusted user starts using the application, he has to decide whether he wants to use it to send location dependent queries (LDQ) or independent location queries (LIQ). In the former case, the user is at home and he can create a task which entails:

- Selecting destination / region (valid region)
- Deciding the QoS that he needs.
- Deciding on a time spam.

This will automatically update the servers (it will store in bit vector, the example below will clarify this point) and save information in the cache (if the device has enough storage). In the latter case, the user who is now onsite (not at home) can implement a task that will lead him to step3.

Another feature which we consider of value is the addition of a "local server prediction list". A prediction list is automatically updated on the local servers which will support some functions such as expanding bandwidth by asking for another base station. This might increase the communication cost but will provide for a better QoS that the user needs.

Step three: Because the user previously requested information which is now in the local servers and part of it could be available as cache on his device, the user can start implementing the tasks associated with this information.

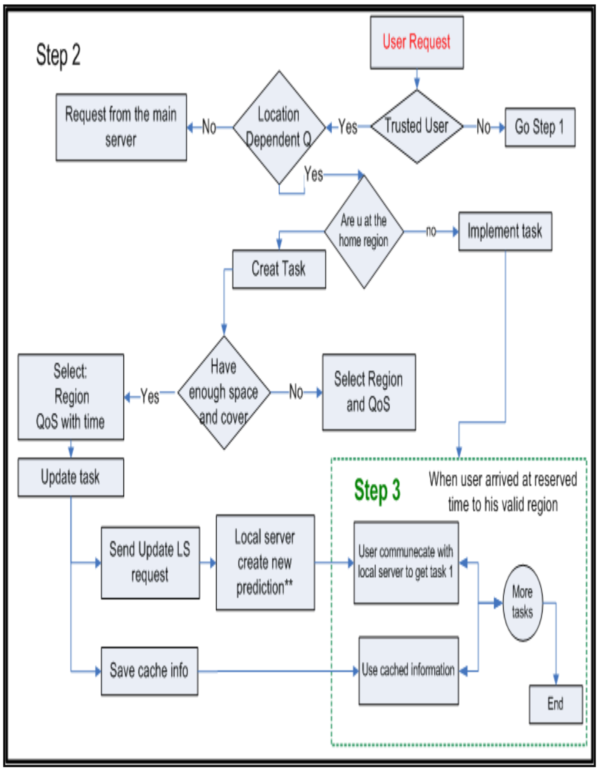

Figure 3. Step 2 and 3 flowchart

A. Example:

Here we extend the bit victor tree to include the presetting process with QoS that the user needs. “Fig. 4” shows an example of many mobile users who created a presetting as follow:

- User 1 (u1): request web browsing (Google map) → the QoS= 00
- User 2 (u2): request video streaming → the QoS= 10
- User 3 (u3): request Virtual class → the QoS=01
- User 4 (u4): request a real time mission with (U5) → both of them has QoS=11.

In the previous example users request different requests in two regions. Their presetting was stored (non-spatial data) in BQR*-tree. For example, user number three (U3) at region two needs to attend a virtual class, so his non- spatial data is =01 the value of the QoS that he needs, where the full value in his leaf node will be (1000********, 1000*******,01).

For more clarification here, we present a pseudo code of the proposed algorithm that supports the presetting technique with QoS.

Object query (OQ), range query (RQ), location independent data from a central server (LIDO), location dependent data.

From the local server (LDDO), MSD main server database are abbreviations that used in this algorithm:

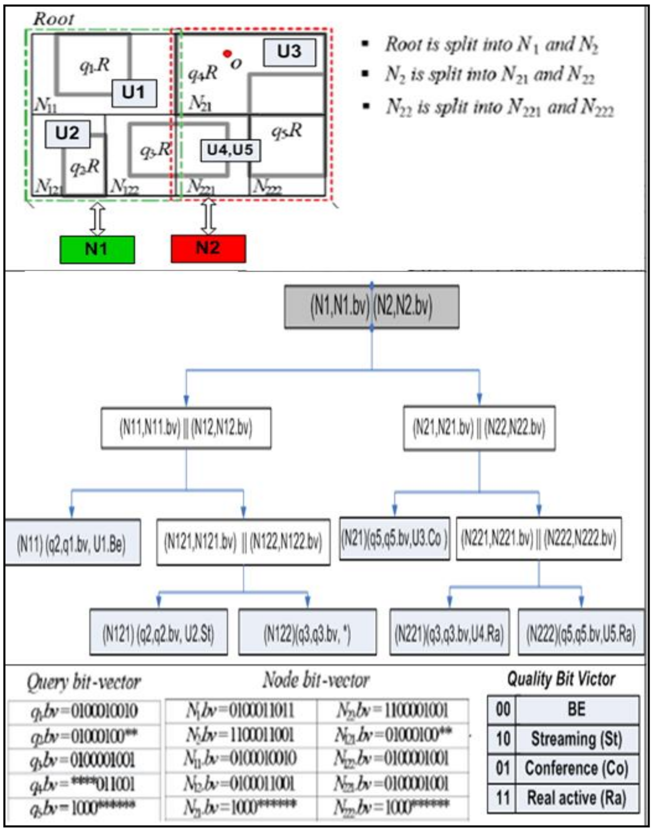

Figure 4. Example of many mobile users who created a presetting

Start : QoS Priority = {1= UGC, 2= ertPS, 3= rtPS, 4=BE}
Pre-setting “Cache”:
{ If (new user)
{ Down load the mobile application;
Get the authentication information; }
Else
{Select the region;
Select the QoS that you need;
Determine span time
Create BQR*-tree.
User-Info = BQR*-tree node-ID, QoS-Priority, time.
Call Local server update(); }
Local server update:
_ MSD ← new user || new task
_ MSD ← Edit and update task;
_ User at site and communicate with User 2.
User at home
_ MSD ←Create task();
_ MSD ←Update task();
User at site:
Send a request to the local mobile application then :
- If the request for an OQ and LDDO, then analyzes the request and looking in the cache.
- Else (not found), then
  _ Send request to the LS (analyzes Q) then decides where to forward the service request:
  _ If (OQ || RQ) && (LIDO) then the request is directly forwarded to the central sever.

End

The system utilizes the GPS for immediate location detection of the user without the actual user's demand. As such, it will be imperative that the mobile phone being utilized backs GPS applications and that the user should be within an open space. Moreover, if there is no network coverage the system depends on the downloaded data "cache" which was downloaded on the mobile phone when the system was installed. In case that network coverage is available, the original data stored with more options will be available and the updates will be more accurate.

Because this type of application is supposed to offer a high quality of service by providing more storage area, multi servers will be used (local server in each location) that will enhance the system performance especially when we distribute the applications on many servers instead of one location with heavy load on one server.

Here we present a simple experiment that demonstrates that the idea of using the QoS in WiMAX topology with distributed servers in many locations (Sinario1) is more effective than WLAN topology (Scenario 2) in "Fig. 5", then "Fig. 6" shows that the WiMAX load is less than WLAN.

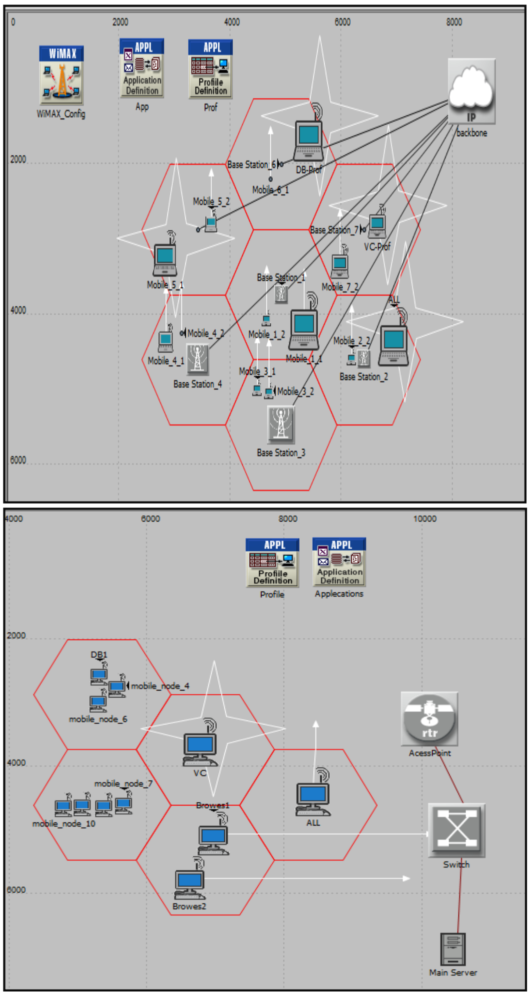

Figure 5. Scenario1– WiMAX configuration that supports QoS
Scenario2 – WLAN without QoS

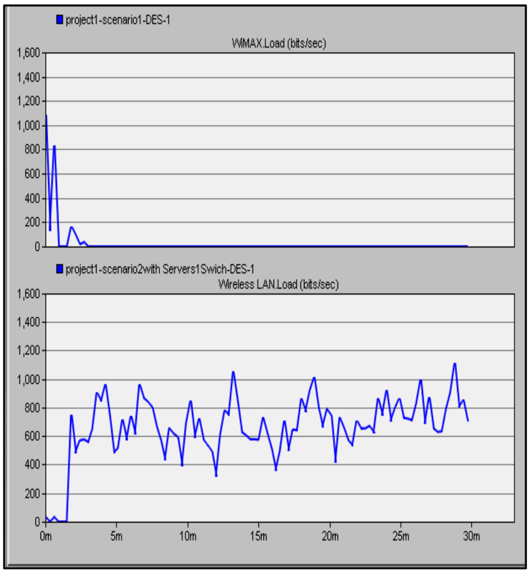

Figure 6. The difference load between WiMAX and WLAN technology

V. Discussion

Nowadays, it has become a necessity that mobile users are provided with quality services. It is essential for any mobile user to retrieve enough information that will help him take decisions in specific situations by providing him with the required non-spatial information about a specific location.

A presetting architecture and algorithm could be one appropriate method to provide spatial and non-spatial data to mobile users with lesser cost (communication or computation). Thus, a combination of presetting and bit-vector tree structure will serve to increase QoS achieved through less computation. The presetting decreases computation of a valid region that the user needs, so neither the user device nor local server continually calculate the valid region. In addition, with so many local servers used, the bandwidth of main server will be low.

We can say that less computation will take place due to a three reasons:

1. Local server: Instead of a central server, local servers help to reduce the network bandwidth cost. Many studies provided an idea that exploit the proxy server as a local server.
2. Presetting: By pre-selecting a region where the user will be available, its continuous calculations for valid regions will be less. In addition, there would less communication with local server.
3. Bit victor tree structure (BQR*-tree): that helps with strong indexing schema help in static spatial query processing that requires less storage.

Opnet simulator shows that WiMAX and IEEE 802.16 are appropriate for transmissions needing to be covered over long distances, a large quantity of nodes, and varying channel bandwidths, which all enable it to be more adaptable so as to providesupport to the various types of data. The technology in question, of wireless solutions utilized for outdoor links, has dramatically enriched in recent years. IEEE 802.16 is aiming to overcome the 802.11 confines in outdoor solicitations by utilizing a deterministic medium access layer. By this, we aim to ensure a significant quality of service, and it is also a key feature for voice applications.

The limitations of our proposed solution can be enhanced through future work, such as Opnet, with limitations for providing a specific data structure or the database features in general. As such, we used an application service with high load database without any details.

Moreover, we will work to evaluate the process on BQR*-tree such as add, edit or delete node. For instance, in cases of evaluation, if the user needs just a location, simple text message or just query about object position, the predicted cost could be off-putting: The user can just use the GPS and the device feature.

VI. Conclusion

Because modern man is almost wholly dependent on mobile applications and services, it becomes a necessity that he is provided with mobile services that require less computations and communication processes, especially when sending search queries. Spatial databases that use range search queries is one application which needs lesser communication processes between server and mobile user.

Major research has been associated with forming productive LBS applications that many fields will benefit from with an instant focus on high QoS. Additionally, LBS servers are able to reply to spatial queries in an efficient manner and time frame while utilizing BQR*tree- like index structures.

Therefore, using many local servers with a presetting selection feature and caching could increase the cost, but achieve high QoS that some users/applications need.

References

[1] Desai. S., "Role Of Information Communication Technologies In Education, Proceedings of the 4th National Conference; Computing For Nation Development, Retrieved from http://www.bvicam.ac.in/news/INDIACom%202010%20Proceedings/papers/Group3/INDIACom10_110_Paper%20(1).pdf , 2010.

[2] Kim, T. "Planning For Knowledge Cities In Ubiquitous Technology Spaces: Opportunities and Challenges". InYigitcanlar, T., Velibeyoglu, K and Baum, S (Eds.). Book: Creative Urban Regions: Harnessing Urban Technologies to Support Knowledge City Initiatives. 2008

[3] Papagiannakis, G., G. Singh, et al., "A survey of mobile and wireless technologies for augmented reality systems, Retrieved Fromhttp://www.miralab.ch/repository/papers/486.pdf, 2007.

[4] Mulloni, A., Dünser, A., Schmalstieg, D. "Zooming Interfaces for Augmented Reality Browsers". To appear in: MobileHCI 2010, Lisbon, Portugal, September 2010.

[5] Naif Aljabri and Fathy Eassa, "Scheduling Manager for Mobile Cloud Using Multi-Agents " International Journal of Computer and Information Technology (ISSN: 2279 – 0764) Volume 02– Issue 03, May 2013

[6] B. Zheng, J. Xu, and D.L. Lee, "Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments," IEEE Trans. Computers, vol. 15, no. 10, pp. 1141- 1153, Oct. 2002.

[7] B. Zheng and D.L. Lee, "Processing Location-Dependent Queries in a Multi-Cell Wireless Environment," Proc. Second ACM Int'l Workshop Data Eng. for Wireless and Mobile Access, 2001.

[8] K.C. Lee, J. Schiffman, B. Zheng, and W.-C. Lee, "Valid Scope Computation for Location-Dependent Spatial Query in Mobile Broadcast Environments," Proc. 17th ACM Conf. Information and Knowledge Management, pp. 1231-1240, 2008.

[9] K.C.K. Lee, W.-C. Lee, H.V. Leong, B. Unger, and B. Zheng, "Efficient Valid Scope for Location-Dependent Spatial Queries in Mobile Environments," J. Software, vol. 5, no. 2, pp. 133-145, Feb. 2010.

[10] B. Zheng, J. Xu, W.-C. Lee, and D.L. Lee, "On Semantic Caching and Query Scheduling for Mobile Nearest-Neighbor Search," Wireless Networks, vol. 10, no. 6, pp. 653-664, Dec. 2004.

[11] Abadi, D.J., Carney, D., Cetintemel, U., Cherniack, M., Convey, C., Lee, S., Stonebraker, M., Tatbul, N., Zdonik, S.B.: Aurora: A new model and architecture for data stream management. VLDB J. 12(2), 120–139, 2003.

[12] Chandrasekaran, S., Cooper, O., Deshpande, A., Franklin, M.J., Hellerstein, J.M., Hong, W., Krishnamurthy, S., Madden, S., Raman, V., Reiss, F., Shah, M.A.: TelegraphCQ: Continuous dataflow processing for an uncertain world. In: Proceedings of the International Conference on Innovative Data Systems Research, CIDR, 2003.

[13] Mohamed F. Mokbel · Walid G. Aref, "SOLE: scalable on-line execution of continuous queries on spatio-temporal data streams", Published online: 20 April 2007, © Springer-Verlag 2007

[14] Norbert Beckmann, Hans-Peter begel,Ralf Schneider, Bernhard Seeger, "The R*-tree: An Efficient and Robust Access Method for Points and Rectangles+ Praktuche Informatlk, Umversltaet Bremen, D-2800 Bremen 33, West Germany.

[15] HaRim Jung a, Yong Sung Kim b, Yon Dohn Chung b, "QR-tree: An efficient and scalable method for evaluation of continuous range queries" 2014

[16] Ilarri, S., Mena, E., Illarramendi, A. "Location-Dependent Query Processing:Where We Are and Where We Are Heading", ACM Computing Surveys 42(3), 1–73 (2010).

[17] H. Hu, J. Xu, D.L. Lee, A generic framework for monitoring continuous spatial queries over moving objects, in: Proc. ofACM SIGMOD'05, 2005, pp. 479–490.

[18] S. Prabhakar, Y. Xia, W.G. Aref, S. Hambrusch, Query indexing and velocity constrained indexing: scalable techniques for continuous queries on moving objects, IEEE Trans. Comput. 51 (10) (2002) 1124–1140.

[19] Y. Cai, K.A. Hua, G. Cao, T. Xu, "Real-time processing of range-monitoring queries in heterogeneous mobile databases", IEEE Trans. Mobile Comput. 5 (7) (2006) 931–942.

[20] Jung, H., Kim, Y.S., Chung, Y.D., "SPQI: An Efficient Index for Continuous Range Queries in Mobile Environments:, Journal of Information Science and Engineering 29(3), 557–568, 2013.

[21] AL-Khalidi H, Abbas Z, Safar M, "Approximate range query processing in spatial network databases". Multimed Syst :1–11, 2012.

[22] X. Gao and A. Hurson, "Location Dependent Query Proxy," Proc. ACM Int'l Symp. Applied Computing, pp. 1120-1124, 2005.

[23] X. Gao, J. Sustersic, and A.R. Hurson, "Window Query Processing with Proxy Cache," Proc. Seventh IEEE Int'l Conf. Mobile Data Management, 2006.

[24] B. Zheng, J. Xu, and D.L. Lee, "Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments," IEEE Trans. Computers, vol. 15, no. 10, pp. 1141- 1153, Oct. 2002.

[25] B. Zheng and D.L. Lee, "Processing Location-Dependent Queries in a Multi-Cell Wireless Environment," Proc. Second ACM Int'l Workshop Data Eng. for Wireless and Mobile Access, 2001.

[26] Haidar AL-Khalidi, David Taniar, Maytham Safar, "Approximate algorithms for static and continuous range queries in mobile navigation "m Published online: 18 October , 2012, Springer-Verlag Wien 2012

[27] Chakchai So-In, Raj Jain, Abdel-Karim Tamimi, "Scheduling in IEEE 802.16e Mobile WiMAX Networks: Key Issues and a Survey" , IEEE Journal on Selected Areas In Communications, Vol. 27, No. 2, February 2009.

[28] J. Zhang, M. Zhu, D. Papadias, Y. Tao, and D.L. Lee, "Location- Based Spatial Queries," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 443-454, 2003.

[29] K.C.K. Lee, W.-C. Lee, H.V. Leong, B. Unger, and B. Zheng, "Efficient Valid Scope for Location-Dependent Spatial Queries in Mobile Environments," J. Software, vol. 5, no. 2, pp. 133-145, Feb. 2010.

[30] S. Nutanong, R. Zhang, E. Tanin, and L. Kulik, "The V_*Diagram: A Query-Dependent Method for Moving NN Queries," Proc. VLDB Conf., pp. 1095-1106, 2008.

[31] Jiun-Long Huang, Chen-Che Huang, "A Proxy-Based Approach to Continuous Location-Based Spatial Queries in Mobile Environments", IEEE transactions on knowledge and data engineering, 2013.

[32] Muhamed Ilyas and R.Vijayakumar, "A Proxy Based Dynamic Data Management using Hierarchical Database for Location Based Services. P School of Computer Science, MG University, Kottayam, Kerala, India, International Conference and Workshop on Emerging Trends in Technology (ICWET), 2010.

[33] Zypad WL 1000, posted Jun. 29, 2007 [online], [retrieved Sep. 22, 2014]. Retrieved from Internet, http://www.pocket-lint.com/news/81360-zypad-wl1000-futuristic-wearable-computer

[34] Hana Al-nuaim, Maiad almasre, "The Use of Augmented Reality as Assistive Services for Pilgrims" Faculty of Computing and Information Technology- Computer Science, Master thesis,2011

[35] Mohamed F. Mokbel · Walid G. Aref, "SOLE: scalable on-line execution of continuous queries on spatio-temporal data streams" The VLDB Journal (2008) 17:971–995, © Springer-Verlag 2007.

[36] Chakka,V.P., Everspaugh, A., Patel, J.M.: Indexing large trajectory data sets with SETI. In: Proceedings of the International Conference on Innovative Data Systems Research, CIDR (2003).

[37] Mokbel, M.F., Xiong, X., Aref,W.G.: SINA: scalable incremental processing of continuous queries in spatio-temporal databases. In: Proceedings of theACMInternationalConference on Management of Data, SIGMOD, pp. 443–454 (2004).

[38] Lee, M.L., Hsu, W., Jensen, C.S., Cui, B., Teo, K.L.: Supporting frequent updates inR-trees: a bottom-up approach. In: Proceedings of the International Conference on Very Large Data Bases, VLDB, 2003.

[39] Kalashnikov, D.V., Prabhakar, S., Hambrusch, S.E.: Main memory evaluation of monitoring queries over moving objects. Distrib. Parallel Databases 15(2), 117–135 (2004).

[40] HaRim Jung, Seongkyu Kim, Joon-Min Gil, and Ung-Mo Kim, " Processing Continuous Range Queries with Non-spatial Selections" , Mobile, Ubiquitous, and Intelligent Computing, Lecture Notes in Electrical Engineering 274, DOI: 10.1007/978-3-642-40675-1_6, © Springer-Verlag Berlin Heidelberg 2014

[41] Andrews, J., Ghosh, A., & Muhamed, R., "Fundamentals of WiMAX Understanding Broadband Wireless Networking"., 2007.