International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2494-9150 Special Issue-01, MAR 2016.

# GEO-SPATIAL DATABASE

[1]Prof. Vishal Shinde, [2]Manisha Durge, [3]Amruta Waghchaude

[1]Asst. Professor, [2,3]BE Student, [1,2,3]Comp. Engg. Dept, SSJCET, Asangaon, India.

[1]mailme.vishalshinde@rediffmail.com, [2]manisha.durge08@gmail.com, [3]amrutawaghchaude2011@gmail.com

**Abstract - The Geospatial database is the new technology in database systems which allow to store and maintain the data. The flow of Geospatial database is starts with design of data model in conceptual, logical and physical data model and then design has been implemented data into object-relational database. A spatial data first preference the query ranks of object based on quality in their neighborhood. In daily life, need to be finding the best/shortest distance between two ways. For example, find the shortest distance way to nearest hospital.**

**Keywords-** ***Spatial network, Database, Web based, shortest path, spatial database, location-dependent and sensitive***

## I. INTRODUCTION

A spatial database manages multidimensional objects, and it provides fast access to those objects based on different selection criterion. The importance of spatial databases is reflected by the modeling entities in a geometric way. For example, locations of hotels, hospitals, rivers and so on are represented as a points in the map, while larger extents such as parks, and landscapes. Many functionalities of a spatial database are useful in various ways. Spatial database systems are also managing large collections of geographic entities, which apart of spatial attributes contain non spatial information (e.g., name, size etc.). The type of interesting preference query, which select the best location with respect to the quality of amenity in its spatial nearest neighborhood. Optimal route query processing finds all the possible routes or ways and then optimizes these possible routes in all of them. For that, the route queries operate on the spatial data with specific information.

The goal is to find the optimal/best route from the given queries. The performances of queries on spatial networks have been an application that is of most interest to the spatial database community. A Google Maps is such an example which has to an interest in replying to queries such as finding nearest neighbor node (e.g. restaurant, rivers) from a set. The objects usually move only on a pre-defined set as specified by the underlying network. Thus, the measure distance is an important in the spatial network application. However, this work is mostly considers in a spaces of spatial query, where distance between two objects is determined by their relative positions. Though some of existing work process query in spatial network, they suffer the problems in the network of expensive for searching cost or storage cost.

The objective of this project is to design a new method to process K-nearest neighbor node in spatial network with minimum storage cost. This is challenging since spatial network with maximum nodes and edges they may result in huge storage or searching cost. They propose a hierarchical algorithm when overcome these problem. The novelty lies in two aspects: (1) The hierarchical method to deal with KNN query in spatial network (2) Decide the proper degree to achieve minimum storage cost while the query is answered in real time. Thus the key contribution is processing K-Nearest node spatial query in real time while minimize the searching cost or storage cost.

The optimal route queries find the entire optimal route from the given set of information. The users have to give a query as a starting point and some travelling constraints or rules along with the database which contains the specific information about the road map. They have various techniques are used for

 

ISSN : 2494-9150 Special Issue-01, MAR 2016.

the processing of route queries. The given constraints may be either total order or the partial order. Optimal route query processes finds all the possible or best routes and then optimize the possible routes. For that, route queries operate on spatial data with specific information. The goal is to find optimal route from the given entire queries.

**In short spatial database is**

- A spatial database system is a database management system.
- It offers to spatial data types in its data model.
- It supports the spatial data types in its implementation and provides at least spatial indexing and efficient algorithms for spatial join.

## II. LITERATURE SURVEY

Evaluate the Top-k Queries over Web-Access Databases. The Optimal Aggregation Algorithms for Middleware. Supporting Top-k join Queries in Relational Databases of the road networks. Compute the Top-t most visited spatial sites. Progressive Computation of the Minimum-Distance of Optimal-Location Query. The Evaluation of All nearest Neighbor Queries.

***Aim and objective***

Aim is based on the network structure of the road to provide security, increase the performance and provide location related to the current location of the user i.e. the user location is updated frequently whenever the user moves from one place to another, hence it display different result in different location based on the current location of user.

The main idea is how the user identity it's exact and protected location of the nearest object. This technique provides security by sending the node and edge information of the user. The nearest destination location of the user queried object can be provided with the additional information like directions.

## III. EXISTING SYSTEM

**Table 1: Comparison of Existing approach**

| | Storage Cost (High/Mid /Low) | Time for Query Processing (Fast/Mid/ Slow) | Real Time Response(Yes/No) | Real Use for Large Network (Yes/ No) |
|---|---|---|---|---|
| Solution based approach | High | Fast | Yes | No |
| Dijkstra based approach | Low | Slow | No | No |
| Pre-Compute based approach | Mid/High | Mid | Yes | Yes/No |
| Pre-Knowledge based approach | Mid | Fast | Yes | No |

## IV. *PROPOSED SYSTEM*

This technique is based on the distance. The location of the user is updated when the user moves. The User sends the query to proxy server. The query is sent from proxy server to the location server along with the k user location [2]. Hence the performance is increased by storing the details of query with respect to the distance. The distance between the current location of user and nearest distance is calculated. If any users fire the same query it will returns the location of the queried object immediately without calculating again. In this technique, the user details are not stored, hence it provides the privacy.

 

### *a. SYSTEM ARCHITECTURE*

Proposed architecture can be shown as follows:

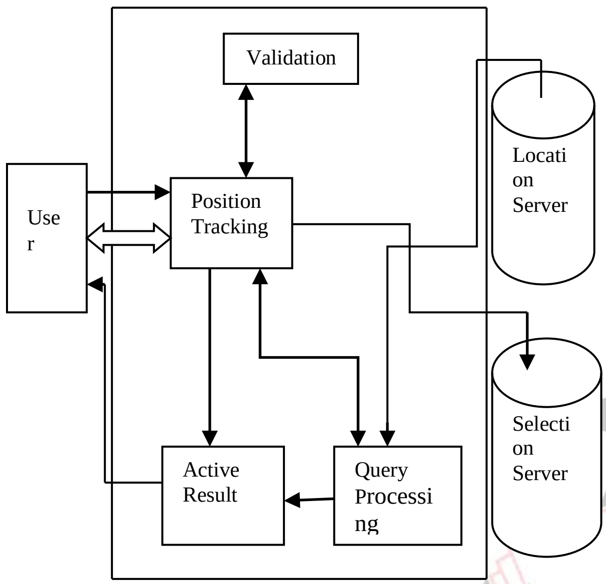


**Fig 1: System Architecture**

The above Fig. explains the architecture diagram of the system. The database is updated frequently when the user moves from one place to another. Here line segments or segments are considered for spatial region. Proxy server calculates the distance between two edges and then sends to the location server. Query processing in the location server results of candidate set to the anonymizer and validated. Active result from the proxy server is returned to the user which contains nearest node and distance to that node from current location.

The nearest way of the user is identified as shown in the fig. The best way stored in the location is searched server based on the traversal based orderings. The breadth search and depth search algorithm are used to traverse from source to destination. It is easy to accept because each and every node is visited at least once. Each and every node is to be visited multiple times. The user order is assigned based on traversal. When the user moves from one location to another then older user location is updated with the new location to identify the current edge.

### *b. Algorithms*

A. *Time Dependent Fastest Path (TDFP)($q,k,t_q$)*

1. // S: Set of nodes, q: query location, dt: departure-time from q
2. // tt: travelling time of the easiest path, $v_i$ : last node added to S
3. // NN: array of current nearest neighbors
4. Initialize S={ q}, t(q)=0, l(v)= ∞ for all v ∉ S
5. $v_i = q$
6. while S ≠ V do
7. for each e ($v_i$,$v_j$) Є E where $v_j$ ∉ S
8. l ($v_j$) = min (l ($v_j$) , $f_e$ (t ($v_i$)))
9. Let w ∉ S such that l (w) = $\min_{v_j \notin S}$ l ($v_j$)
10. S U {w},t(w) =l(w), $v_i$=w
11. If $v_i$ is a data object Then
12. add $v_i$ to NN;
13. tt = t($v_i$) – tq; // compute travel-time
14. End If
15. If NN size() = k
16. Then break;
17. end while
18. return NN and travel times

## V. MATHEMATICS MODEL

Given a set of points P and a group of queries Q , an aggregate K nearest neighbor query is to find a set of points R that contains K objects such that for any $p \in (P - R)$ and any $p' \in R$ ,dist(p',Q)≤dist(p,Q)) where for any object $o$ $dist(o,Q) = \sum_{\forall q_i \in Q} dist(o,q_i)$

As an example, consider the few friends are wanted to meet at some place so that the overall distance travelled is minimized. In this case the friends are the query group and the places are data points. An aggregate nearest neighbor query will return the place or city so the total distance travelled by all of them in minimum. Aggregate nearest neighbor queries can also be used for clustering and outlier detection. They give another interesting and simple application of aggregate nearest neighbor queries. The speed of a very large circuit depends on the relative distance between various components, the aggregate nearest neighbor queries can be issued for find the

 

ISSN : 2494-9150 Special Issue-01, MAR 2016.

abnormalities and guide relocation of components. They present three algorithms to answer of aggregate nearest neighbor queries called *Multiple Query Method* (MQM), *Minimum Bounding Method* (MBM) and *Single Point Method* (SPM). All these algorithms travel from R-tree to answer Aggregate Nearest Neighbor (ANN) queries. Since their experiments show that MBM performs the other two algorithms.

To prune the search space, MBM uses minimum bounding rectangle M that contains all query points in Q. The algorithms traverse R-tree and prune the nodes that cannot contain any user point. The prune nodes based on two observations. Let point M be the MBR of the point Q, and best_dist be the distance of the best aggregate nearest neighbor found. A node $N$ can be pruned if mindist(N,M)≥best_dist/n where mindist(N,M) is the minimum distance between nodes N and M and $n$ is the cardinality of Q . Consider the example, where the node N1 can be pruned because mindist(N1,M)=3>(best_dist/2=2.5).

This pruning can also be applied on the data points. Whenever the data point P is retrieved, first its minimum distance from the MBR, M is calculated and the data point P can be pruned if mindist(p,Q)>best_dist/n.

The observation provides a fixed pruning bound. Any node N can be pruned if $\sum_{\forall q_i \in Q} mindist(N, q_i) \geq best\_dist$

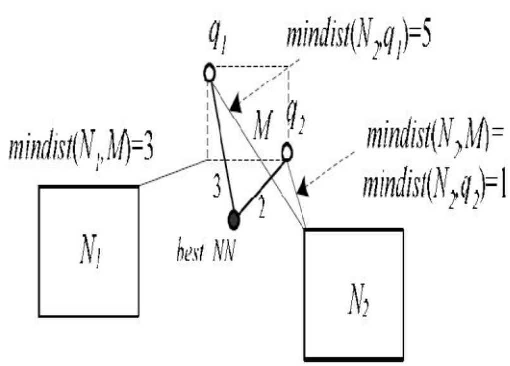


***Fig2:-*** **Minimum Distance**

where mindist(N,$q_i$) is the minimum distance between the node N and a query point . $q_i$ In the example of Fig.7.1 the node N2 is pruned because mindist($N_2$,$q_1$)+mindist($N_2$,$q_2$)=6>best_dist, propose two ellipse-based methods for the aggregate nearest neighbor nodes and queries. Solve the problem of aggregate nearest neighbors on the road network. The aim is to find minimize the total sum of the distances to the query points from a point p .introduce two other variants of ANN queries. In first variant the aim is to find a point pεP with the smallest distance from any query point in Q.

## VI. EXPECTED OUTPUT

In our experiment, the road network is created as graph image with nodes and edges. The users are move on the road and destination locations are marked in the road network. Our algorithm is implemented in ASP .Net as front end, and the node information are stored using SQL server. The mouse click event is used to stop the user movement. One of the users is stopped to raise a query. For example If the user wants to reach the nearest hospital or hotels, then the nearest hospital or hotel location with the distance and the location area is displayed using our algorithm. Single query should be processed at a time. Storing large quantities of information and providing access to selective data through queries in an efficient way. They manage information in a single central repository.

First select the option what you want, hotel or hospital. Then click on Go Button, displayed the list of hotels or hospitals with address and distance.

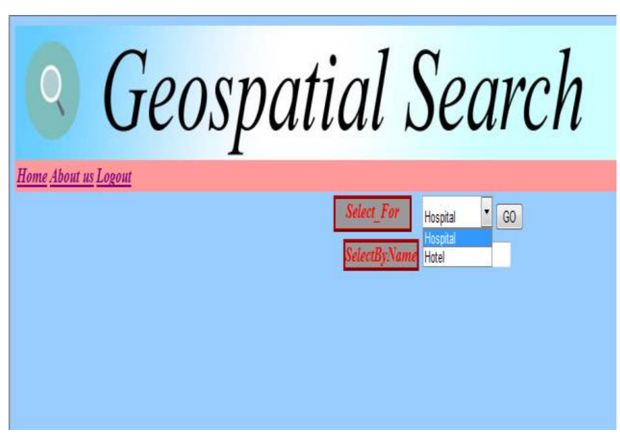


**Fig 3: Searching Process**

ISSN : 2494-9150 Special Issue-01, MAR 2016.


**Fig 4: Expected output**

## VI. CONCLUSION

Google map and geo spatial data is becoming more popular and people use them frequently. Search nearest destination and required places or buildings. Also road mapping is more frequently used for finding the road directions and proper path.

In all those things the spatial database plays important part and helps to sort the problems and find new things.

## REFERENCES

[1] Yufei Tao and Cheng Sheng,"Fast Nearest Neighbor Search with Keywords", Vol. 26, pp. 878-888, April 2014.

[2] David Walkie, Jason Sewall, and Ming C. Lin, "Transforming GIS Data into Functional road Models for Large-Scale Traffic Simulation", Vol. 18, pp. 890-901, June 2012.

[3] Rathinaeswari. S.P,R. Parvathi, "Enhanced Network Based Query Processing in Road networks" VOL.2, pp. 65-72, April 2013.

[4] Marius Muja, Member, IEEE and David G. Lowe, Member, IEEE "Scalable Nearest Neighbor Algorithms for High Dimensional Data",VOL.36,No.11,pp. 2227-2240,November 2014.

[5] Man Lung Yiu, Nikos Mamoulis, and D. Papadias,"Aggregate Nearest Neighbor Queries in Road Networks",VOL.17,NO.6,pp.820-833,June 2005.

[6] S. Shekhar ,D. Liu, "CCAM: A Connectivity-Clustered Access Method For Network and Network Computations",VOL.19,NO.1,pp.102-119,January 1997.

[7] I. Balaban, "An optimal Algorithm for Finding Segments Intersections", pp.211-219, 1995.

[8] J.Sewall, D. Wilkie, P. Merrell,and M. Lin, "Continuum Traffic Simulation", Vol. 29, pp. 439-448,2010.

 