Extending SQL for Moving Objects Databases

Kwang Woo Nam, Jai Ho Lee, Min Soo Kim
4S Integration Technology Research Team, Spatial Information Technology Center, ETRI

161, Kajong-Dong, Yuseong-Gu, Daejeon 305-350, KOREA
TEL : +82-42-860-6760
FAX : +82-42-860-4844
E-mail: {kwnam, snoopy, minsoo}@etri.re.kr

Abstract

This paper describes a framework for extending GIS databases to support moving object data type and query language. The rapid progress of wireless communications, positioning systems, and mobile computing devices have led location-aware applications to be essential components for commercial and indurstrial systems. Location-aware applications require GIS databases system to represent moving objects and to support querying on the motion properties of objects. For example, fleet management applications may require storage of information about moving vehicles. Also, advanced CRM(Customer Relationship Management) applications may require to store and query the trajectories of mobile phone users. In this trend, maintaining consistent information about the location of continuously moving objects and processing motion-specific queries is challenging problem. We formally define a data model and query language for mobile objects that includes complex evolving spatial structure, and propose core algebra to process the moving object query language. Main profit of proposed moving objects query language and algebra is that proposed model can be constructed on the top of GIS databases.

Keywords : Moving Objects Databases, LBS, GIS

1. Introduction

A characteristic of real-world objects is that their state in space changes over time. As computing power and technology grows, new advanced applications manage such space and time varying objects, such as land parcel, rivers, roads, taxis, buses, fishing boats, air planes, cars, and cellular phone users, etc. In applications for these objects, the current state as well as the past and anticipated future states of the objects are frequently of interest[Eriw99, Jens01, Seyd01].

Time varying spatial objects may be classified into two types. One is discrete spatiotemporal objects, which can be stored as a sequence of each state in time, such as history of land parcel information in land information system. This brings about the need for spatiotemporal databases. During last a decade, spatiotemporal databases have been a active research field. Another is continuously moving objects, such as taxi position in tracking system. This brings moving objects databases. Moving objects databases is beginning as background engine of Location-based Service(LBS) applications. Recent advances in wireless networking, location determination technologies, and mobile applications. have led to the emergence of LBS[Sh00]. LBS is a growing technology field that focuses on providing GIS and spatial information via mobile and field units. Also, Combined

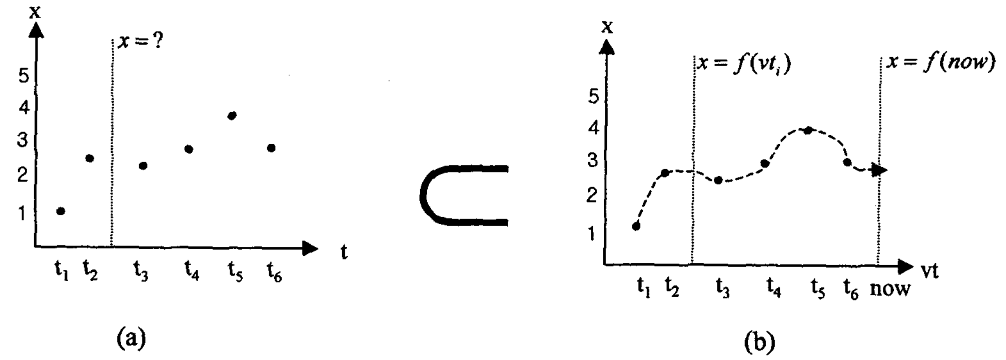

Figure 1. Representation of a moving point: (a) Database storage representation (b) Abstract Representation

with the massive adoption of pervasive computing devices, they enable an important class of mobile commerce applications, such as location-sensitive billing and queries, targeted advertisement, and retail store navigation.

Although the former researches on moving objects databases focus on the general case of geometries that may change in a continuous manner, one should note that there is a class of applications where geometries change only in discrete steps. In this paper, we propose a moving objects data model and moving objects SQL with time-series and continuous moving object operators. This is distinguished from other moving objects researches by Guting, Wolfson, and others.

The rest of this paper is organized as follows. In section 2, we introduce related work with moving objects data model and query language. In section 3, we describe characteristics of moving objects, and propose new time-series operators. In section 4, we extend SQL to support moving objects databases. Lastly, in section 5 we offer conclusions.

2. Related Works

Guting have developed a data type oriented approach for modeling and querying moving objects. The idea is to consider the two major abstractions moving point and moving region as abstract data types with suitable operations that can be embedded into a DBMS data model and query language as attribute types[Erwi99, Forl00].

Also, The group of Wolfson has proposed a concept of moving objects databases that is complementary to Guting[Sist97, Wolf98]. Whereas Guting's approach of modeling describes movement in the past, hence the complete history of moving objects, their focus is on capturing the current movement of entities, e.g. vehicles, and their anticipated locations in the near future. The basic idea is to store in a database not the actual location of an object, which would need to be updated frequently, but instead a motion vector describing location, speed and direction for a recent instant of time. As long as the predicted position based on the motion vector does not deviate from the actual position more than some threshold, no update to the database is necessary. They also offer a query language based on temporal logic to fomulate questions about the near future movement. This approach is restricted to moving objects, not applied to moving regions.

Proposed moving objects data model and query language is distinguished from Guting's and Wolfson's woks in the point of time-series operators. They neglect the time-series characteristics of moving objects and have focused on continuous moving objects characteristics in spite that time-series characteristic is

$validtime(e_i) = t_i$	
$first(s,k) = e_k$	$index(e_i) = i$
$last(s,k) = e_k$	$next(e_i) = e_{i+1}$
$count(s) = l$	$previous(e_i) = e_{i-1}$
$min(s,c_i) = v$	$max(s,c_i) = u$
$after(s,t_i) = e_j$	$slice\_squence(s,k_{from},k_{to}) = s'$
$before(s,t_i) = e_j$	$slice\_sequence(s,t_{from},t_{to}) = s'$

(a) Time-series operators

snapshot	$snapshot\_validtime(m,t_i) = v_i$
	$snapshot\_value(m,c_i,x) \subseteq domain(Time)$
slice	$slice\_validtime(m,t_{from},t_{to}) = m'$
	$slice\_value(m,c_i,x) = m'$
project	$project\_validtime(m) \subseteq domain(Time)$
	$project\_value(m,c_i) \subseteq domain(T_i)$

(b) Moving objects operators

Figure 2 Operators for moving objects

important and useful in location-based applications.

3. Moving Objects Data Model

In this section, we propose a new model for moving objects. This supports time-series operators as well as continuous moving objects operators.

3.1 Characteristics of Moving Objects

In real world, moving objects occupy a part of Euclidian space over varying time. But, in computerized world, since managing all of such information is impossible, we may manage it in the abstracted world which is transformed from real world. For example, Storing all information about the trajectory of cellular phone users into computer storage is impossible as well as inefficient. However, the trajectory information is gathered in every sampling time as frequent as capable, and stored in database storage.

Figure 1 shows representations of moving point for cellular phone users. Suppose that a phone user is walking through a road. This information will be stored into database as a sequence of positions with time tag like Figure 1(a). This is similar with characteristic of time-series data. In spite that this is very useful to evaluate variation of moving objects and manage the data sequentially, other researches of moving objects databases have neglected. A problem of database storage representation is that we can get the exact position in sampling time t_x, but not in any time between sampling time t_x and t_{x+1}. For querying continuous moving objects,, it should be estimated by a mapping function having time parameter in abstract representation as shown Figure 1(b)

3.2 Time-Series View

As described in section 3.1, moving objects have characteristics of time-series data. We define a element of moving objects as following.

$$e = < t \,||\, o >$$

A element is a tuple with time and value object. For example, first value of Figure 1(a) is $<t_1||1>$. A moving object is a sequence of such elements.

$$s = [e_1, e_2, ..., e_{now}]$$

We propose time-series operators for moving objects as shown Figure 2(a). Validtime operator returns a time t of the element, and value operator return a value object o of the element. First operator return first element e_1 of the moving object, and last operator return last element e_{now}.

Aafter operator returns the nearest element after the time t in parameter, and before operator returns the nearest element before time t. Slice_sequence operators return the part of sequence restricted by times or indexes in parameters.

3.3 Continuous Moving Objects View

For continuous moving objects view, we propose three kinds of operators, snapshot, slice, and project. Snapshot operators returns a snapshot by the specific dimension described in parameter. These consist of snapshot_validtime and snapshot_value operator. Snapshot_validtime operator return snapshot value object in specific time point. Oppositely, snapshot_value return snapshot times by value dimension. Slice operators return sliced moving objects by the specific dimension described in parameters. Slice_validtime operator return a moving object between specific time period. Oppersitely, slice_value operator return moving objects sliced value objects. Project operators return value objects projected by another dimension, time or value.

Slice operators are distinguished from snapshot operators in returned values. The former return moving objects, but the latter return snapshot objects in value dimension.

4. Moving Objects SQL

In this section, we extend SQL to support moving objects databases. This extended SQL uses time-series and moving objects operators described in section 3.

4.1 Creation of a Moving Objects Table

Moving objects are managed as data types, such as mpoint, mlinestring, and mpolygon. For example, when a user want to create a table for managing positions of moving cellular phone in location-based service applicatons, he may create a table with attribute of moving point data type as shown example 1.

Example 1. Create a gmlc_user table with a moving point attribute.

```
CREATE TABLE gmlc_user (
        cellphone_number     char(20),
        cellphone_owner      char(15),
        position             mpoint
         option (Data Split Mode is time 'every 1 day'
                   index is 3DRTree)
);
```

Data split mode means how to clustering method of moving point datum. Data split mode consist of three kinds, such as time, spatial, and storage. In upper example, it would be clustered every 1 day. Index mode means index type for moving point. In this example, position attribute will be managed by 3DRTree.

4.2 Time-Series Queries

Time-series operators described in section 3.1 can be used in SQL for querying moving point, such as getting the latest input position, partial sequence, and aggregating variation among variation of movements. We will show the efficiency through the following examples.

Example 2. Show the latest and count of a user's position information, whose cellular phone number is 016-402-8040.

```
SELECT last(position, 0), count(position)
FROM gmlc_user
WHERE cellphone_number = '016-402-8040';
```

Example 3. Show moving points during one hour from the time tx.

```
SELECT slice_sequence(position, :tx,
            :tx+ interval('1 hour'))
FROM gmlc_user
```

4.3 Continuous Moving Objects Queries

Continuous moving objects queries consists of snapshot, slice, project, and join queries. As described in section 3.3, Snapshot operator can be used for getting a snapshot by a specific time. We show a use case of snapshot operator in example 4.

Example 4. Show the current positions of gmlc_users.

```
SELECT snapshot_validtime(position, now)
FROM gmlc_user
```

Example 5. Show users within a specific area at current time..

```
SELECT cellphone_owner, cellphone_number
FROM gmlc_user
WHERE within(snapshot_validtime(position, now),
polygon('3454.32 3423.43...'));
```

In upper examples, we used a time variable 'now' which means the latest value of cellular phone user since the database can take the exact position in current time. If the user need, explicit time value can be used instead of 'now' variable for getting a snapshot in the past.

Slice queries are used the history information of moving objects. The followings are examples for slice queries.

Example 6. Show trajectories of gmlc users during specific period.

```
SELECT slice_validtime( position,
            period('2002/05/01-2002/05/31'))
FROM gmlc_user
```

Example 6 shows that slice_validtime operator is used for select phase. This restrict the trajectories within user specific range.

Example 7. Show gmlc users who pass the specific area during

```
SELECT cellphone_owner
FROM gmlc_user
WHERE overlaps(position, motpolygon(
  period('2002/05/03-now'),polygon('34.3 22. ...'))
```

Function for moving objects can be used in SQL, such as distance. Distance operator is very useful to extract the spatial relationship between two moving objects. Query as like example 8 will be efficient in location-based service application for moving objects databases.

Example 8. Show gmlc users who less than 100m distance from another user.

```
SELECT A.cellphone_owner, B.cellphone_owner
FROM gmlc_user A, gmlc_user B
WHERE DISTANCE(A.position, B.position) < 100
```

5. Conclusion

In this paper, we proposed moving objects operators and moving objects SQL. Proposed model and query language support time-series view as well as continuous moving objects view. Specially time-series view is firstly proposed in this paper.

These characteristics can be used in moving objects database applications for location-based services. In fact, we have implemented the moving objects database with proposed characteristics. In implementation phase, we will extend to support more powerful operator for moving objects.

References

[Cote01] J.A. Cotelo Lema, Luca Forlizzi, Ralf H. Guting, Enrico Nardelli, Markus Schneider, "Algorithms for Moving Objects Databases", FernUniversitat Hagen, Informatik-Report 289, October 2001.

[Erwi99] M. Erwig, R. H. Guiting, M. Scheneider, and M. Vazirgiannis, "Spatio-Temporal Data Types : An Approach to Modeling and Querying Moving Obejct in Databases," GeoInfomatica, Vol.3, No.3, pp.269-296, 1999.

[Forl00] L. Forlizzi, R. H. Guting, E. Nardelli, and M. Schneider, "A Data Model and Data Structures for Moving Objects Databases," ACM SIGMOD Conference, pp.319-330, 2000.

[Jens01] C. S. Jensen, A. Friis-Christensen, T. B. Pedersen, D. Pfoser, S. Saltenis, and N. Tryfona, ``Location-Based Services - A Database Perspective," Proceedings of the Eighth Scandinavian Research Conference on Geographical Information Science, As, Norway, June 25-27, 2001, pp. 59-68.

[Seyd01] A. Y. Seydim, M. H. Dunham and V. Kumar, "Location dependent query processing," Second

ACM international workshop on Data engineering for wireless and mobile access, pp.47 - 53, May 20, 2001, Santa Barbara, CA USA

[Sh00] S. L. Sh, D. Wa, "Handling Disaggregate Spatiotemporal Travel Data in GIS," GeoInformatica Vol.4, No.2, pp.161-178, 2000.

[Sist97] A. P. Sistla, O. Wolfson, S. Chamberlain, S. Dao, "Modeling and Querying Moving Obejcts," ICDE, pp.422-432, 1997.

[Wolf 98] O. Wolfson, B. X., S. Chamberlain, L. Jiang, "Moving Objects Databases: Issues and Solutions", Proc. of the 10th Int. Conf. on Scientific and Statistical Database Management (SSDBM98), Capri, Italy, July 1-3, 1998, pp. 111-122.