Advanced Science and Technology Letters
Vol.45 (CCA 2014), pp.60-64
http://dx.doi.org/10.14257/astl.2014.45.12

Design Considerations on Implementing an Indoor Moving Objects Management System

Qian Wang, Qianyuan Li, Na Wang, Peiquan Jin

School of Computer Science and Technology,
University of Science and Technology of China, 230027, Hefei, China

Abstract. In this paper, we present a framework to implement a database management extension for indoor moving objects based on a commercial Object-Relational DBMS. As most people are living in indoor spaces, it is valuable to investigate people's moving trajectories in indoor space so as to develop indoor location-based services. We first present the general idea of managing indoor moving objects via ORDBMS, as well as the detailed design on the data types and operations. Then, we give the implemental results and a case study is presented.

Keywords: Indoor Space; Indoor Moving Objects; DBMS

1 Introduction

Traditional moving object data management focused on GPS-supported moving objects in outdoor environment [1-2], and little effort has been done on moving objects in indoor space. However, people spend most of their time in indoor space, such as office building, shopping mall, metro station and museum. Recently, wireless positioning techniques like RFID, wifi and Bluetooth offer opportunities for us to track indoor moving objects. Thus it brings new challenges in indoor moving objects management [3].

Indoor space has some unique features, compared with outdoor space [4]. Firstly, the moving of objects is constrained by rooms and doors. In particular, objects have to pass by the door when moving from one room to another one. Secondly, the distance measurement is different from that in outdoor space. The latter usually employs the Euler distance. However, this is not applicable in indoor space, due to the existence of doors and rooms. Finally, the positioning ways in indoor space usually use sensors like RFID and Bluetooth, which are differing from the GPS receiver in outdoor environment. The indoor positioning techniques are not able to report the precise positions of indoor moving objects.

Previous research on indoor moving objects management were mainly focused on data models [1], indexes [2], and specific indoor query processing [2], whereas little work has been done in the implementation of real database management systems for indoor moving objects. To our best knowledge, there are no real systems built so far.

Aiming at providing practical support of indoor moving objects management for various commercial applications, we present an extension of Object-Relational

ISSN: 2287-1233 ASTL

DBMS in this paper, which is called IndoorDB. IndoorDB is based on our previous work on indoor data model [4] and is developed using the cartridge technology provided by Oracle, which enables us to add new data types as well as functions and indexing methods into the kernel of Oracle. The unique features of IndoorDB can be summarized as follows:

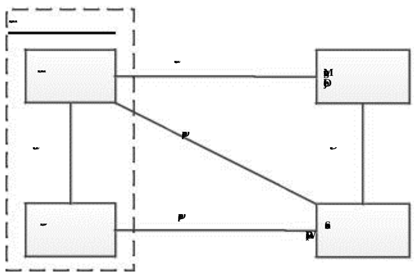
(1) It is SQL-compatible and built on a widely-used commercial DBMS. Thus it can be easily used in real database applications for indoor scenarios and provides a practical solution for indoor moving objects management under current database architecture.

(2) It supports various moving objects types specially designed for indoor LBS applications, such as moving number, moving bool, moving string, indoor position, indoor space, and indoor moving object. Combined with ten types of spatiotemporal operations supported by those new data types, users can represent indoor space and indoor moving objects for various applications. Moreover, different types of spatiotemporal queries are supported by IndoorDB, such as indoor navigation, indoor KNN, trajectory queries, range queries, and so on.

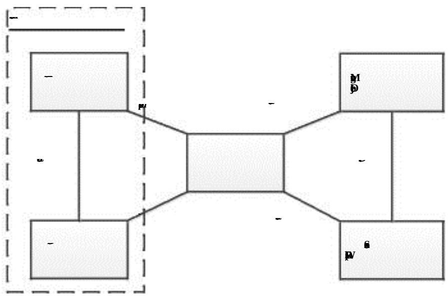
The remaining of the paper is structured as follows. In Section 2, we briefly introduce the design of a data model for indoor space. Section 3 is focused on the data types, which will be extended into ORDBMS. Section 4 presents the implemental issues, and Section 5 concludes the paper.

2 A Data Model for Indoor Space

Indoor space data model needs to describe three kinds of indoor entities, including indoor elements, sensor deployment and moving object data. Fig. 1(a) is the entity-relationship graph of traditional symbolic model. *Indoor element* includes *Room* and *Door*, and *Room* is connected with *Door*; *Moving Objects* are detected by *Sensor*; *Sensor* is deployed in *Room* or *Door*; *Moving Object* is located in *Room*. As indoor element is abstracted as symbol, there is no geometric data of indoor element, which leads to symbolic moving object location and inaccurate indoor distance calculation.



(a) symbolic model



(b) proposed model

Fig. 1. Entity-relationship graph of indoor space model

To address this problem, we propose to add entity *Grid* as indoor space primitive geometry (as shown in Fig. 1(b)). *Room* and *Door* are composed with *Grids*, which can represent geometry information. *Sensor* and *Moving Objects* are located in *Grids*, which are more accurate than symbolic locations.

We design new data types to represent indoor space and indoor moving objects, namely *temporal data types*, *spatial data types*, and *moving objects types* (as shown in Fig.2). The *moving objects types* contain moving base types and indoor moving object types. The former refers to the numeric, Boolean, or string values changing with time, whereas the latter refers to the indoor moving objects as well as their trajectories. All the new data types are implemented by PL/SQL using the CREATE TYPE statement.

In Fig.2, the *indoor spatial types* are defined on the basis of the data model shown in Fig.1, in which we divide the whole indoor space as a set of grid. The sensors and doors are defined as *indoor position* types.

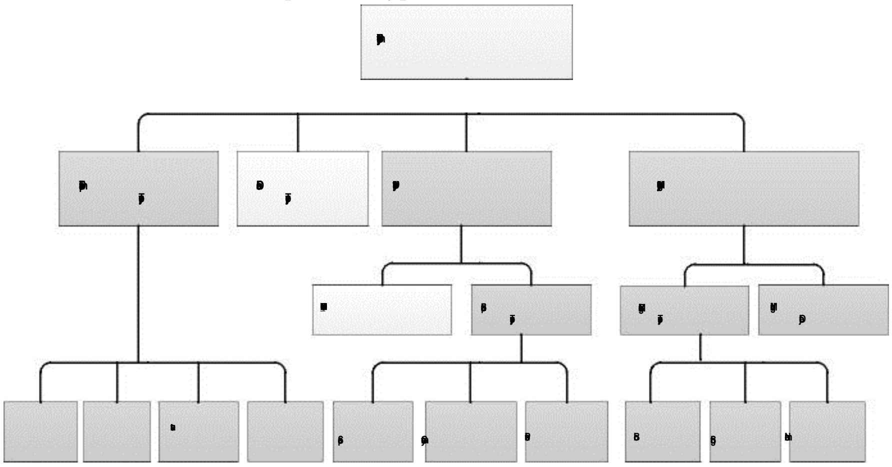


Fig. 2. The type system of IndoorDB

3 Implementation

Instant Period

Indoor
Indoor
Indoor

Moving
Moving
Moving

The IndoorDB data model can be effectively implemented on typical object-relational DBMS. Object-relational DBMS provides extensions support for new data types and functions, and has been studied in traditional spatiotemporal data management [6]. The architecture is shown in Fig. 3. The *C Library* provides the core support for the indoor moving objects data types and operations. The *IndoorDB* module is the component that actually brings the indoor moving objects management support into the DBMS. Once installed, it becomes an integral part of the DBMS; no external modules are necessary. The module is written mostly in C, and makes heavy use of the *C Library*. The *ORDBMS Extending Tool* is provided by the DBMS.

When the *IndoorDB* module is installed into DBMS, users can use standard SQL to gain indoor location-based services from the DBMS. No external works impose on users.

Each new data type (ADT, abstract data type) is implemented and registered into the object-relation DBMS. When creating a new data type, some support routines must be provided, which will deal with the casts between the internal representation in DBMS and the external representation out of DBMS.
Now we can use the new data types to represent indoor space and moving objects. For example, we can create following table to represent a shopping mall as well as the moving objects (people) inside it:

```
Mall(Mid:number,Mname:string,Mbuilding:inmgg);
Stores(Rid:number,Mid:number,Rname:string,Rreg:inregion
);
Doors(Did:number,Mid:number,Dloc:ingrid);
Customers(Cid:number,Cname:string,Ctraj:mov(ingrid))
```

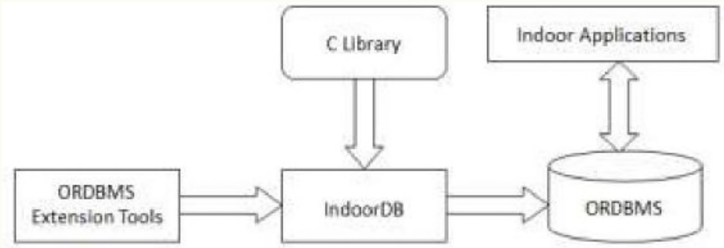


Fig. 3. Implementation architecture for IndoorDB

Due to the missing of real dataset for demonstrating indoor moving objects management, we also develop an indoor data generator called IndoorSTG [7, 8]. Using IndoorSTG, we can generate simulated indoor moving trajectories, which can be further imported into IndoorDB for query evaluation, as shown in Fig.4

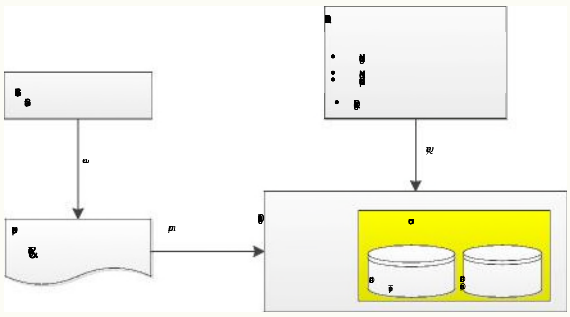


Fig. 4. Scenario of IndoorDB evaluation

4 Conclusion

In this paper, we proposed some design considerations on the data model and implementation of indoor moving objects. We present a preliminary design for a grid-based indoor space model, and define the new data types for representing indoor space as well as indoor moving objects. We have implemented the proposed indoor data model on Oracle using the cartridge extension technique. In the future, we will concentrate on the optimization of the indoor moving objects extension.

Acknowledgement. This paper is supported by the National Science Foundation of China (No. 61379037).

References

1. R. Güting, M. Böhlen, et al. A Foundation for Representing and Querying Moving Objects. TODS, 25(1): 1-42 (2000)
2. R. Güting, V. Almeida, et al., Modeling and Querying Moving Objects in Networks. The VLDB Journal, 15(2): 165-190 (2006)
3. C. Jensen, L. Hua, Y. Bin, Indoor—A New Data Management Frontier. IEEE Data Engineering Bulletin-Special Issue on Spatial and Spatio-temporal Databases, 33(2): 12-17 (2010)
4. P. Jin, L. Zhang, Semantics and Modeling of Indoor Moving Objects, International Journal of Multimedia and Ubiquitous Engineering, 7(2): 153-158 (2012)
5. Q. Li, P Jin, L Zhao, S Wan, L Yue, IndoorDB: Extending Oracle to Support Indoor Moving Objects Management, The 18th International Conference on Database Systems for Advanced Applications (DASFAA'13) , LNCS 7826, Springer, pp. 476-480 (2013)
6. P. Jin, P. Sun, OSTM: A Spatiotemporal Extension to Oracle. In Proc. Of NCM , pp.575-580 (2008)
7. C. Huang, P. Jin, H. Wang, N. Wang, S. Wan, L. Yue, IndoorSTG: A Flexible Tool to Generate Trajectory Data for Indoor Moving Objects, the 14th International Conference on Mobile Data Management (MDM'13) , IEEE CS, Milan, Italy (2013)
8. H. Wang, P. Jin, L. Zhao, L. Zhang, L. Yue, Generating Semantic-Based Trajectories for Indoor Moving Objects, In Proc. Of WAIM Workshops, LNCS 7142, Springer, pp.13-26 (2011)

