Design and Implementation of Advanced Traffic Monitoring System based on Integration of Data Stream Management System and Spatial DBMS

Ying Xia* Hongmei Gan** GyoungBae Kim***

Abstract The real-time traffic data is generated continuous and unbounded stream data type while intelligent transport system (ITS) needs to provide various and high quality services by combining with spatial information. Traditional database techniques in ITS has shortage for processing dynamic real-time stream data and static spatial data simultaneously. In this paper, we design and implement an advanced traffic monitoring system (ATMS) with the integration of existed data stream management system (DSMS) and spatial DBMS using IntraMap. Besides, the developed ATMS can deal with the stream data of DSMS, the trajectory data of relational DBMS, and the spatial data of SDBMS concurrently. The implemented ATMS supports historical and one time query, continuous query and combined query. Application programmer can develop various intelligent services such as moving trajectory tracking, k-nearest neighbor (KNN) query and dynamic intelligent navigation by using components of the ATMS.

Keywords : Traffic Monitoring System, Data Stream Management System, Spatial DBMS, Intelligent Transportation System

1. Introduction

Intelligent transportation system (ITS) [1] applies innovative and advanced technologies such as information processing, sensor network, mobile communications, GPS and auto-control to make transportation systems more efficient and customers service-oriented. Recent developments in network and sensor device technologies enable us to easily obtain real time traffic information by which the advanced traffic monitoring system (ATMS) can be developed to provide real time traffic monitoring[22].

ATMS need to handle stream data and spatial data concurrently because continuous and unbounded stream data of traffic information are arrived from the sensor network and many applications use the spatial data to implement map services. Data stream management system (DSMS) [2,3] efficiently handles data stream and executes query whenever new data stream arrives, while it cannot manages the spatial data. While, spatial database management system (SDBMS)[4] can deal with large volume of static and spatial data storage and query but can not process continuous and unbounded data stream. The existing traffic monitoring prototype system [5,6,7,8] combined with DSMS and SDBMS only involves in stream processing on regular areas and is based on specially developed data stream management systems.

Since the objective of DSMS is real-time processing a large number of arriving data stream from sensors and, on the other hand, that of SDBMS is handling the large volume of stored spatial data in database[23], it is difficult to combine DSMS and SDBMS. In this paper, we design and implement a

* Professor, Chongqing University of Posts and Telecommunications(South-west JiaoTong University), xiaying@cqupt.edu.cn
** Master Student, Chongqing University of Posts and Telecommunications, ganxinfeng@yahoo.com.cn
*** Assistant Proessor, Computer Education Department of Seowon University, gbkim@seowon.ac.kr(corresponding author)

Manuscript received : 2009.02.04 Revised : 2009.04.02 Accepted : 2009.04.13

GIS based ATMS with the integration of existed DSMS and SDBMS using IntraMap [9]. Our developed ATMS can deal with the stream data of DSMS, the trajectory data of relational DBMS, and the spatial data of SDBMS concurrently. And also the implemented ATMS supports three kinds of query: historical one time query, continuous query and combine query. Historical one time query is only involved in the static and historical data that stored in relational DBMS or SDBMS. The continuous query is related with time-varying data stream that process by DSMS. The combine query is processed based on integration of stream data, relational data and spatial data. These queries can serve for various intelligent services such as moving trajectory tracking, k-nearest neighbor (KNN) query and dynamic intelligent navigation.

The remained contents are organized as follows. Related works of SDBMS and DSMS are reviewed in section 2. In section 3, the data types and query models of ATMS are presented. Section 4 highlights the system architecture, proposes data integrating and query execution methods. Then, we show the execution examples and performance evaluation of the implemented ATMS. Finally, section 5 gives the conclusion of this paper.

2. Related works

SDBMS supports spatial data models, abstract data types (ADTs) and query language. It plays an important role for efficient spatial data management in many GIS applications. Several commercial SDBMS have developed and widely used, such as ESRI [10] Arc/View in Canada, ERDAS IMAGINE, MapInfo [11] in America, GEOMania [12] GDK and KSIC's IntraMap in Korea, SuperMap [13] in China. ERDAS Imagine is a GIS product developed by ERDAS, provides the most comprehensive interoperable geospatial solutions available. Arc/View is the world's most popular desktop GIS and mapping software. With Arc/View you can create intelligent, dynamic maps using data from virtually any source and across most popular computing platforms. Arc/View provides data visualization, query, analysis, and integration capabilities along with the ability to create and edit geographic data. MapInfo support address geocoding、site selection、mapping customer/competitor locations, problem notification, emergency response coordination, data visualization and analysis. MapInfo datasets in native (TAB) format and in interchange (MIF/MID) format are supported for reading and writing. Update of existing files is not currently supported. SuperMap is developed by SuperMap company in BeiJing of China, with most functions of GIS software. IntraMap is a high-end geographic information software that supports GIS data management and spatial query tools in both C/S and internet environment, developed by KSIC (Korea geoSpatial Information & Communication Co., Ltd.). It supports visualization, edit data, decision-making, geoprocessing, and so on. The biggest advantage of KSIC's products is that spatial data can in accordance with attribute data across Microsoft Access database.

DSMS [14,15] can process huge and fast data streams from multiple sources, enable real time response in despite of unpredicted system loads. It requires an integrated model on both persistent relations and time-varying data streams to support continuous queries. New processing paradigms and methods have been proposed and implemented in several stream processing systems to achieve the similar objectives. However, they can only handle streaming point locations naively while do not have adequate support to take account of the spatial and temporal information simultaneously.

Here we present some stream processing systems. CarTel [5] is a mobile sensor computing system designed to collect, process, deliver and visualize mobile sensor data. It provides a simple query-oriented programming interface and manages intermittent and variable network connectivity. CarTel is powerful in processing mobile sensor data, but is weakness in combine queries involved in SDBMS and DSMS. STREAM [16] is a general-purpose DSMS that supports a declarative query language and is designed to handle high-volume data streams with large numbers of complex continuous queries. PLACE [6] is a query processor for handling real time spatio-temporal data streams in regular region with a scalable location-aware database server, but it is helpless in the irregular spatio-temporal queries. Nile [17], a query processing engine for data streams, extends the query processor engine of an object-relational database management system, Predator [18], to process continuous queries over data streams. But Nile can't support spatio-temporal queries. KOREDstream [19] is a prototype DSMS developed by the Database Lab of Inha University in Korea who corporately studied spatial DBMS and SDBMS with us.

In this paper the KOREDstream system is used as our SDBMS engine because of its easily accessed through its easy and detail user interface. And the data types and query model which studied in [20] is used for system test data generation and system performance test. This kind of data types and query model is studied in section 3 more detail.

3. Data types and Query model

This section presents the data types and query models of the ATMS for traffic monitoring according to the user requirements in ITS. In the ATMS, stream data and spatial information can be processed simultaneously.

3.1 Data types

There are commonly kinds of data in ATMS such as road networks, car information, sensor data, and so on. The data model is displayed in Figure 1. All of the data can be divided into three types:

1) Spatial data: As the basic data in GIS and ITS, spatial data abstract the entries in real world into geometry model and are stored in SDMBS. Spatial data are mainly presented in map which consists of four layers:

- Road : main elements of the map, it presents the whole road network and be abstracted as lines in the map.
- Car : moving objects on the road, it is abstracted as a point in the map.
- Administrative Region : district is divided according to administrative boundaries and abstracted as polygon in the map.
- Plot : polygon contains other entries such as shopping mall, park, gas station and so on, it is abstracted as polygon in the map.

2) Attribute data: It describes the static attribute information of filtered historical stream data and spatial objects such as roads, cars, districts and surface features. This information is stored in relational DBMS (RDBMS) or SDBMS.

3) Stream data: Data streams are continuously supplied. The data stream s is represented as the sequence of tuples <(a1, t1), (a2, t2), …, (am, tn)>, where am is attribute value of the tuple and tn is the timestamp. These data describe the changing attribute information of objects such as speed and position of a car and processed by DSMS.

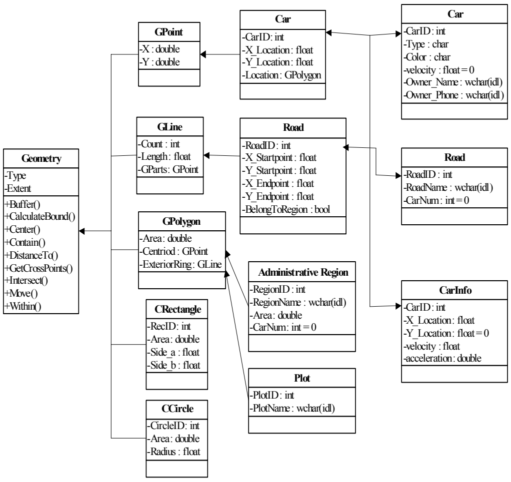


Fig. 1. Data model of ATMS

3.2 Query model

ATMS supports three types of query: historical one time query, continuous queries and combine query, which are proposed in MITRE technical report in 2006[20].

- *Historical one time* query : These queries involve in historical data such as car's trajectory, they are processed once over snapshot of data set.
- *Continuous query* : These queries are produced over time and continuously reflect the data arriving. It may have various forms depending on the mutability of objects and queries, and may ask about the present or future information. For example,

Q1 : Which cars are speeding at present?

Select CarID, Speed, X_location, Y_location, Timestamp
From CarStreamTable
Where Speed > StandardValue;

- *Combine query* : The combine queries in our system are defined as queries involved in relational query, spatial query and stream query. For example,

Q2 : How many taxi are in the NanShan area within the next 5 minutes ?

SELECT count(CarID)
FROM CarStreamTable, Car_Table, District_Table
WHERE StreamDataTable.location in District_Table.area

AND Car_Table.car_type = 'taxi'
AND District_Table.name = NanShan
AND CarStreamTable.Timestamp BETWEEN present_ moment AND present_moment + 5;

4. Implementation of the ATMS

The ATMS integrates DSMS and commercial SDBMS to support various queries for intelligent transportation system.

4.1 The System Architecture

The architecture of ATMS consists of four layers: *ATMS services, ATMS engine, global query manager* and *integrated data manager*. *ATMS services* provide users advanced traffic monitoring services which are implemented by using the combination of various functions of ATMS engine. ATMS Engine is a component-based core part which provides general functions with the support global query manager. *Global query manager* parsing SQL or continuous query language (CQL) and send the data processing requirement to *integrated data manager* which is comprised of DSMS, relational DBMS (RDBMS) and SDBMS. Figure 2 shows architecture of ATMS.

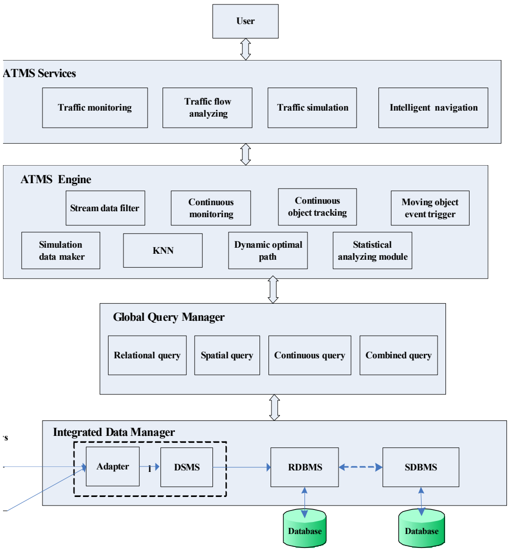


Fig. 2. System architecture of ATMS

Integrated data manager is emphasis on hybrid data management and processing. Because data coming from different streams and their type or fields may be different, it is necessary to create an unbounded relational table for each stream. Each delivery unit from a stream is regarded as tuple with a timestamp in the relation. The last layer of Figure 2 shows that real-time stream data such as position and speed of vehicles are gathered from sensors and sent to adapter. Adapter is a stream process engine. It receives and cleans stream data, then sends them to DSMS at a special speed. If the arrived vehicle record already exists in the table, then just update it, otherwise a new tuple is inserted. Historical data such as trajectories of vehicles are store in RDBMS if necessary. RDBMS and SDBMS can exchange data to each other to support combine query.

Global query manager supports historical one time query, continuous query and combine query with the support of the integrated data manager. As soon as the global query manager receives a query, it checks the feasibility, validates and executes the query plans, finally feeds back to applications.

ATMS Engine is component-based and provides several strategies for real-time monitoring and management. *Stream data Filter* checks stream data that match the given conditions. *Continuous Monitoring* monitors not only the traffic flow of a specific area but also each car's status real-time and continuous. *Continuous objects tracking* tracks the moving objects real time and continuously. *Moving object event trigger* executes pre-defined actions whenever an anomaly event is automatically checked. *KNN* performs constant monitoring of k-nearest neighbor queries over moving cars within a geographic area. *Dynamic optimal path supports dynamic* shortest or best path searching. *Statistical analyzer* using the forecasting methods to analyze the traffic conditions based on the historical data and spatial data. *Simulation data Maker* allows user change the map, adjust the interrelated road sections, or modify the attribute information for traffic simulation.

ATMS services provide advanced traffic monitoring services based on relative components of ATMS Engine. For example, *Traffic Monitoring* provides real time traffic monitoring with the support of moving object event trigger, continuous monitoring, stream data filter, and continuous object tracking components. *Traffic flow Analyzing* plays an important role in traffic prediction and planning based on the stream data filter and statistical analyzing components. *Traffic Simulation* is used to simulate the traffic situation after traffic planning and implemented by continuous monitoring and simulation data Maker components. *Intelligent Navigation* searches for the shortest or best path involving KNN and dynamic optimal path components.

4.2 Query Processing of ATMS

Historical one time query focus on querying from the existing data repositories, static sources data from RDBMS or SDBMS are joined to make a result set. In a standard historical query view, the administrator or client submit a query request that may be related to former multiple data, to achieve this, raw data is combined with information from multiple sources to create a result set on the client display. Figure 3 shows the typical sequence of operation. For example trajectory query "Select CarID, X, Y, TimeStamp From Car Where CarID=52689" on RDBMS in Figure 3, when the global query manager receives the registered query (step 1), it analyses the user request, analyzes queries, performs query optimization, and constructs query plans. Then generates one-shot queries to RDBMS or SDBMS (step 2) and the result is returned to user (step 3).

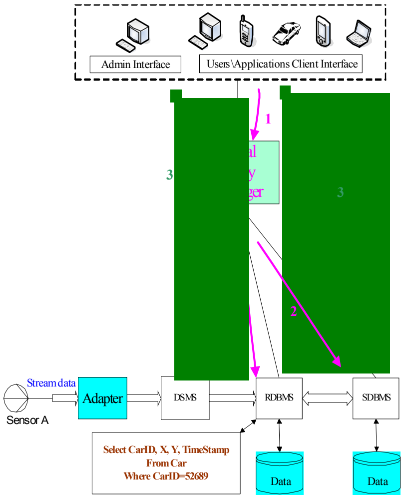


Fig. 3. Historical one time queries processing

Continuous query is activated by anticipated events such as data occurring to accesses dynamic stream data in DSMS. A continuous query is activated by events, such as data arrival and time progress, and it generates differential results using new data that have arrived after the last execution of the query. Continuous queries involved in both current and future results. In this scenario, the query represents a live combination of occurring and anticipated events. Figure 4 depicts a typical transaction flow. The returns for current events are expected to return multiple results over time as the dataset stream continually evolves. With regard to "Q1: Which cars are speeding at present?" in Figure 4, when the global query manager receives the registered query (step 1), it analyses the user request and generates continuous queries to DSMS (step 2). A window could be specified by time intervals in a FROM clause for each data stream. When a continuous query starts, tuples arriving within the window are rapidly returned (step 3).

We can specify a window for each data stream. Windows are defined by time intervals in a FROM clause. When a continuous query is triggered, tuples arriving within the windows are used to generate the results.

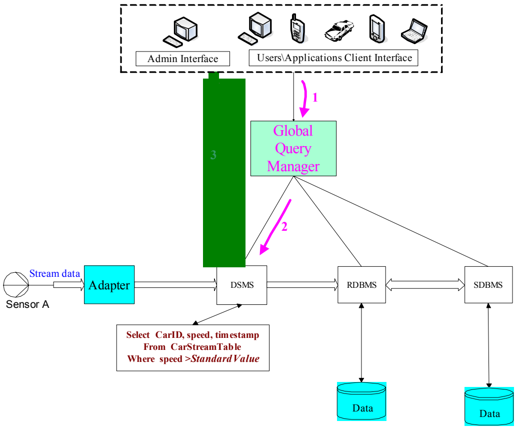


Fig. 4. Continuous queries processing

Combine query refers to query on both dynamic stream data in DSMS and static data in RDBMS or SDBMS. The global query manager analyses the combine query and generates continuous query and relative subqueries. Some results return from RDBMS and SDBMS are regarded as filter conditions or joined together. Finally, integrated data manager return the executed results user. For combined queries, the order of return is not certain. Current events may occur either before or after the results of a historical query are returned. We must filter large numbers of stream data using DSMS for SDBMS while executing some queries that involved in information in DSMS and DBMS. The potential processing of combined queries are depict as follows (Figure 5). For example in Figure 5, “How many taxis are in the NanShan area within the next 5 minutes ? ” when the global query manager receives the registered combine query (step 1), it analyses the user request and generates subqueries to DSMS, RDMBS and SDBMS (step 2). It first inquire about the are of NanShan district on SDBMS, then filter the coming stream on DSMS for 5 minutes according to the X, Y coordinate to judge whether cars are in the NanShan area or not. Finally, it checks car type using gathered the CarID from relational DBMS and shows the results of the combine query on the map by using functions of spatial DBMS (step 3, 4).

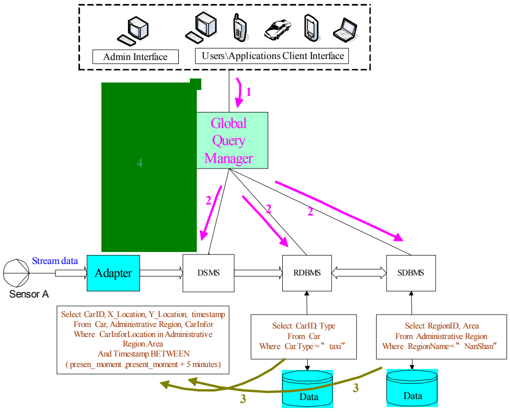
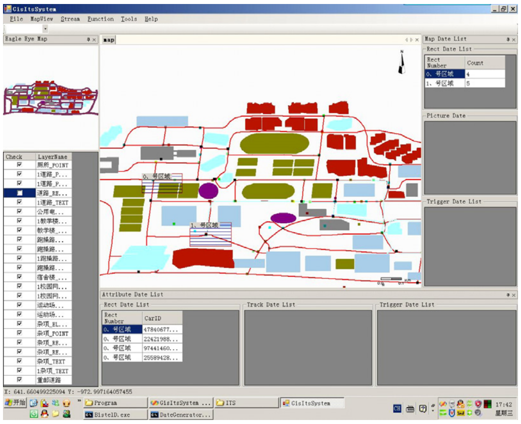


Fig. 5. Combine queries processing

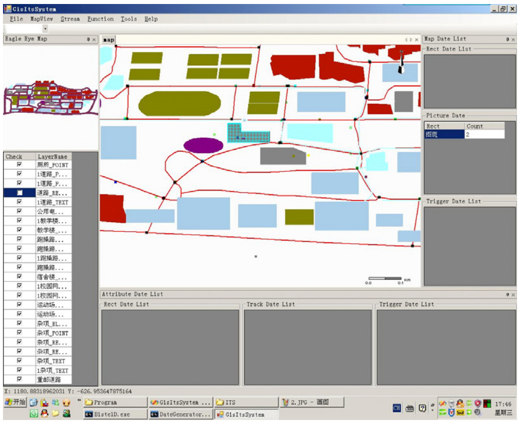
4.3 Case Study and Performance Evaluation

In the prototype system, stream data is generated by ITS Data Generator (DG). The DG generates traffic stream data such as CarID, speed, acceleration and location, considers traffic regulations including car-following model, lane-changing model and traffic lights control model. Car-following model adopts the improved PARAMICS (PARallel MICroscopic Simulator) Car-following model, proposed a dynamic safety distance according to the Leutzbach's static safety distance. Lane-changing and traffic lights model use sample traffic control model. The DG is visualized using GIS and the traffic electronic map adopts campus map of CQUPT in ChongQing (China). All experiments were performed on a 2.8 GHz Pentium 4 PC with 1.0 GB main memory, running Microsoft XP Professional SP2. All methods are implemented using C# and C++. RDBMS is Oracle 10g, DSMS is KOREDstream, spatial data management and GIS implementing use IntraMap/Objects.

Figure 6 demonstrates a monitoring service of ATMS. The real-time traffic information is displayed at center, while map control and service information are shown at both sides. Figure 6a) is example of Q1: *Which cars are speeding at present?* The number of speeding cars in each area and the detail car information are listed. Figure 6b) is an example of Q2: “*How many cars are in the NanShan area within the next 5 minutes ?*” The selected plot is highlighted in the map and the information table of PlotID and numbers of vehicles are shown on the right.



(a) Example query 1(Q1)



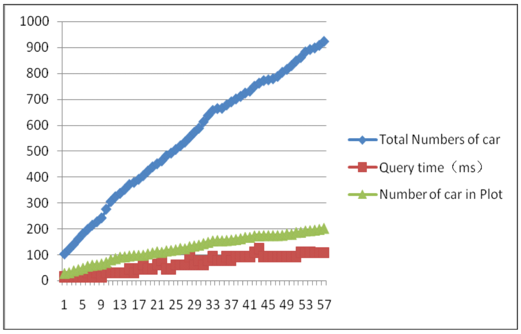
(b) Example query 2(Q2)

Fig. 6. Execution examples of ATMS

In figure 7, we present preliminary performance evaluations of ATMS that include the results of continuous query and combine query. We use the DG to generate different number of vehicles that move along the road network of a given city. Horizontal axis represents time sequences and the vertical axis represents numbers. According to experimentation, ATMS can execute Q1 and Q2 within 200ms respectively.

6. Conclusions

This paper has presented some of challenges that must be meet by the next generation of ATMS. We have cast these research problems in context of the ATMS design and implementation in order to draw out the issues and to experience how to combine the existed systems.



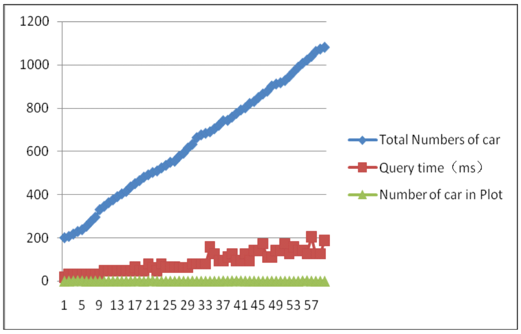


Fig. 7. Performance evaluation of ATMS for Q1 and Q2

In this paper, we propose a prototype system of a GIS based real-time traffic monitoring, the developed ATMS can deal with the stream data of DSMS, the trajectory data of relational DBMS, and the spatial data of SDBMS concurrently. It also supports history one time query, continuous query and combine query based on the integration of existing database productions. From the view of data management, dynamic traffic stream data are processed in DSMS, filtered historical data are stored in the relational DBMS, traffic related spatial information are organized in SDBMS. Global query processing scheme analyses the feasibility and generates the query plan, experiment shows the validation of the prototype system. In the future, different combination of stream data manager and relational database can be tested for real time traffic monitoring, and spatial-temporal mining algorithms can be executed and optimized based on this data platform.

References

[1] Monahan, Torin. “War Rooms” of the Street: Surveillance Practices in Transportation Control Centers. The Communication Review 10 (4): pp.367-389, 2007.

[2] Kitagawa, H, Watanabe, Y. “Stream Data Management Based on Integration of a Stream Processing Engine and Databases,” NPCW2007, 2007.

[3] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. “Models and issues in data streams,” PODS2002, 2002.

[4] Ralf Hartmut G?ting, An Introduction to Spatial Database Systems, VLDB Journal, vol.3, 1994.

[5] Bret Hull, Vladimir Bychkovsky, Yang Zhang, Kevin Chen, Michel Goraczko, Allen Miu, Eugene Shih, Hari Balakrishnan, and Samuel Madden. “CarTel: a distributed mobile sensor computing system,” SenSys2006, 2006.

[6] M. F. Mokbel, X. Xiong, M. A. Hammad, and W. G. Aref. “Continuous Query Processing of Spatio-temporal Data Streams in PLACE,” STDBM2004, 2004.

[7] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. “Traffic View: A Scalable Traffic Monitoring System,” MDM2004, 2004.

[8] Yamada, S, Watanabe, Y,Kitagawa, H.; Amagasa, T. “Location-Based Information Delivery Using Stream Processing Engine Mobile Data Management,” MDM 2006, 2006.

[9] 한국공간정보통신, IntraMap, http://www.ksic.net/ksic/product/product_02.asp

[10] ESRI, http://www.esri.com

[11] MapInfo, http://www.mapinfo.com

[12] GEOMania, http://www.geomania.com

[13] SuperMap, http://www.supermap.com

[14] D. J. Abadi, et al. "The Design of the Borealis Stream Processing Engine," CIDR2005, 2005.

[15] D. Abadi, D. Carney, U. Cetintemel, M. Cherniack, C. Convey, S. Lee, M. Stonebraker, N. Tatbul, and S. Zdonik. "Aurora: A New Model and Architecture for Data Stream Management," VLDB Journal, vol. 12(2), August 2003.

[16] A. Arasu, B. Babcock, S. Babu, J. Cieslewicz, M. Datar, K. Ito, R. Motwani, U. Srivastava, and J. Widom. "STREAM: The Stanford Data Stream Management System," 2004.

[17] M. A. Hammad, M. F. Mokbel, M. H. Ali, W. G. Aref, A. C. Catlin, A. K. Elmagarmid, M. Eltabakh, M. G. Elfeky, T. M. Ghanem, R. Gwadera, I. F. Ilyas, M. Marzouk, and X. Xiong. "Nile: A Query Processing Engine for Data Streams," ICDE2004, 2004.

[18] P. Seshadri. "Predator: A resource for database research," SIGMOD Record, 27(1):16-20, 1998.

[19] Jae-Syn Shin, Byeong-Seop You, Sang-Hun Eo, Dong-Wook Lee, Hae-Young Bae. "Data Stream Storing Techniques for Supporting Hybrid Query," Journal of Korea Multimedia Society, Vol.10, No.11, 2007, pp.1384-1397.

[20] Don Landing, Mark Munson, Stephan Nadeau, Seth Landsman, Brigit Schroeder, Eddy Cheung, Virgil Zetterlind, Dan Potter. "Sensor Data Analysis Framework Research Roadmap," MITRE technical report, 2006.

[21] Peng Chunhua, Yan Lei, Liu Jianye, Zheng Jianghua, He Lina. "The Theoretical Research on the New Car-Following Model for PARAMICS," The Second Seminar of Intelligent Transportation System Forum, 2005.

[22] Yang Koo Lee, Keun Ho Ryu, "Historical Sensor Data Management using Temporal Information," Journal of Korea Spatial Information System Society, Vol.10, No.4, 2008.

[23] Joung Joon Kim, Pan Gyu Kim, Dong Oh Kim, Ki Young Lee, Ki Joon Han, "Design and Implementaion of RTLS based on a Spatial DSMS," Journal of Korea Spatial Information System Society, Vol.10, No.4, 2008.

Xia Ying

1986~1989 Electronic Science Dept., ChongQing Univ. of Post and Telecom (B.S.)
1999~2001 Inha University(M.S.)
2006~Present Doctoral Student, Computer Application Dept., South-west JiaoTong University
2003~Present Prof. of Computer department of ChongQing Univ. of Post and Telecom., PR China
2006~Present ChongQing Univ. of Post and Telecom., PR China, Director of Sino-Korea GIS Research Center
Research Interests : Database, GIS, LSB Policy and Technology, Data Stream

Gan Hong Mei

2003~2008 GIS Dept., ChongQing Univ. of Post and Telecom(B.S.)
2008~Present Computer Science Dept.,, ChongQing Univ. of Post and Telecom (M.S.)
Research Interests : Database, GIS, Data Stream

Gyoung Bae, Kim

1988~1992 Dept. of Computer Science and Engineering, Inha University(B.S.)
1992-1994 Dept. of Computer Science and Engineering, Inha University(M.S.)
1994~2000 Dept. of Computer Science and Engineering, Inha University(Ph.D.)
2000~2004 Senior Researcher, ETRI.
2004~Present Assistant Prof., Dept. of Computer Education, Seowon University
Research Interests include : Moving Objects & Real-time Database, Storage System, GIS, VOD