GML Map Visualization on Mobile Devices

Eun-Ha Song* and Young-Sik Jeong*

Abstract—GIS can only be applied to certain areas by storing format. It is subordinate to a system when displaying geographic information data. It is therefore inevitable for GIS to use GML that supports efficient usage of various geographic information data and interoperability for integration and sharing. The paper constructs VisualGML that translates currently-used geographic information such as DXF (Drawing Exchange Format), DWG (DraWinG), or SHP (Shapefile) into GML format for visualization. VisualGML constructs an integrated map pre-process module, which filters geographic information data according to its tag and properties, to provide the flexibility of a mobile device. VisualGML also provides two major GIS services for the user and administrator. It can enable visualizing location search. This is applied with a 3-Layer POI structure for the user. It has trace monitoring visualization through moving information of mobile devices for the administrator.

Keywords—Map Visualization, DXF, DWG, SHP, GML, POI, Trace Monitoring

1. INTRODUCTION

Mobile technology is rapidly emerging as a prominent icon representing current technology. The need for a geographic information service is rapidly rising due to propagation of wireless PDA, car navigation systems, and cell phones. Current GIS were developed with multiple methods and forms to display, analyze, process, store, and gain geographic information data. There exist shortcomings in terms of free usage, integration, application, and sharing due to the limited format of GIS. Therefore, a common interchangeable format is necessary. OGC (Open GIS Consortium) has suggested a GML (Geography Markup Language) [1] specification to save and transfer geographic information containing a geographic aspect that has spatial or non-spatial attributes. GML gives a structure to data allowing flexibility of access to information to provide interoperability of GIS [2, 3].

The paper develops VisualGML. VisualGML visualizes and creates common-format GML that covers heterogeneity of geographic information data, supporting its interoperability. VisualGML designs IMP (Integrated Map Pre-processor) to overcome insufficient data process memory and low connection speed caused by visualizing geographic information data on a mobile device. IMP lightens geographic information data by extracting unnecessary information of file formats, such as DXF [4], DWG, or SHP [5], and categorizing these into layers. VisualGML user-centric visualization is based on hierarchical POI information, rather than on simple GML-based map visualization. It also saves, gains, traces, and manages real-time movement information of a mobile device.

※ This paper was supported by Wonkwang University in 2009.
Manuscript received January 13, 2010; revised February 8, 2010; accepted February 16, 2010.
Corresponding Author: Young-Sik Jeong
* Department of Computer Engineering, Wonkwang University, Korea({ehsong, ysjeong}@wku.ac.kr)

2. RELATED WORKS

2.1 Map representation based on GML

Existing GML-base map visualization systems include: TatukGIS Viewer 1.4 [6] by Tatuk-GIS Inc., FME Universal Viewer [7] by SafeSoftware Inc., Master Map Viewer 2.0 [8] by Snowflake Software Ltd., iSMART Explorer 4.4 [9] by eSpatial Inc., and GML Viewer by ETRI.

iSMART Explorer 4.4 is easy to use, has a light-weight application, and is connected to the OCI (Overseas Consultants Incorporated) DB; thus, being much easier to analyze. It finds the GML scheme automatically and re-examines, expands, minimizes, and moves non-spatial property data. It also allows sufficient spatial editing through the web browser, and provides fast response time for low bandwidth connection. TatukGIS Viewer 1.4 supports a few file formats that could process up to 2Gbyte files. It also translates visualized geographic information into a PDF file, and measures the distance and size between two points. In addition, with vector property information it displays identical properties of a map with the same color, and when clicking a vector graphic with a mouse it shows a property window. Similarly, FME Universal Viewer 2004 displays identical properties with the same color, shows a property window; additionally, it allows editing of geometrical properties.

While most other map visualization systems besides programs support simple visualization, they cannot provide user-centric map information. They are not very useful as they are offline, single applications. VisualGML uses various existing geographic information, extracts core elements for the map display, and converts them into GML. Thus, VisualGML supports heterogeneity and interoperability of geographic information data. It provides hierarchical location search and map visualization applied with hierarchical POI DB during the process of GML conversion. Moreover, VisualGML gains and saves movement coordinates of a mobile device, and provides a tracking monitoring visualization.

2.2 The existing monitoring systems

Existing monitoring systems are divided into ones with visual tracing and ones with only raw data in tables. A typical example of the former is the Tracing Monitoring Solution of Cyber Map World (CyberMapWorld Co. Ltd.) [10]. It is a kind of tailored solution for tracing through a GIS system to offer different applications the detailed information in real-time. Such applications include the movement course and location information of the vehicles, customers to enterprises, government and public offices that need to monitor vehicles, distribution, and customers. The system is composed of the LBS engine to manage the mobile objects with large capability and the GIS engine of the technology applied with the main memory DB. It is a system to supply a solution based on the LBS server including the detailed DXF files with a reduced scale. It can trace the mobile objects on an extensive scale and all the vehicles and the users, and offer an exceptional search speed out. The NEO control system, if the call is registered in the window of user's devices, indicates the call lists before the allocation, and after allocating the vehicles; the passenger's state is shown in the allocation list. If you enter the number of the car to trace, this system will show the latest location information of it based on DXF geographic information data. The managers can control the refreshing intervals and frequency of the data based on the states of the system. However, the majority of existing trace monitoring systems described above do not provide standard geographical information, because their data is based on the existing numerical map, DXF format. Furthermore, none of them can support and provide the integrated location information for those mobile device users. In this paper, we propose a system to trace and manage the mobile devices based on the international recommended GML standards to overcome the above limitations.

3. VisualGML System

3.1 VisualGML Architecture and Control Flow

VisualGML comprises of a map server and mobile device. The map server manages the lowered-weight geographic information data, generating the GML file, and the tracking monitoring visualization of a mobile device. The mobile device is concerned with a user-centric event process and visualization. Fig. 1 overviews the VisualGML architecture.

The map server performs the map data process that a mobile device requests, mainly through Service Broker. Service Broker receives location coordinates and the map area of a mobile device from the Communication Module, orders GML generation, and requests monitoring visualization. GML Maker generates GML using IMP, which accepts various map formats and extracts core elements from them for map display, and POI DB, which provides location information to the users. Monitoring Manager visualizes GML generated by GML Maker onto the Monitoring Viewer. A map is initially visualized without location tracking information of a mobile device. However, tracking monitoring information is visualized when requested by the manager.

The mobile device requests the map and mediates visualization through the View Frustum Manager. After the mobile device receives location coordinates from the GPS, the GPS Coordinate Transformation Module converts the coordinates into the TM coordinate system. The Communication Module transfers location coordinates and map area requested. View Frustum Manager transfers the received GML file through the GML Parser. Parsing GML, GML Parser extracts the map property, and sends visualization via View Broker. View Broker mediates the real-time map visualization under location coordinates change, and manages the event process due to user-altered events, such as expand, minimize, or move.

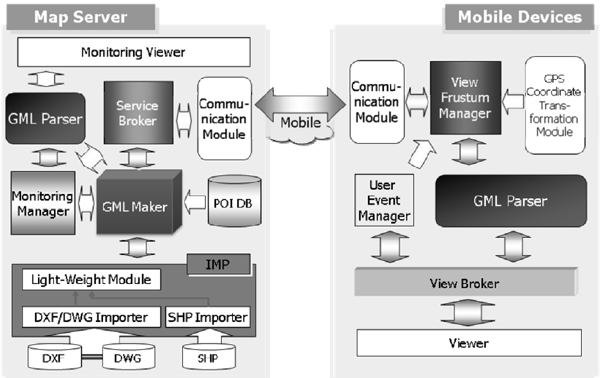

Fig. 1. VisualGML Architecture

3.2 Integrated Map Pre-processor

Improved performance of visualizing the data onto the mobile device is required due to the massive volume of geographic information data. File formats, such as DXF, DWG, and SHP, that are currently used widely are visualized onto the specific application. This paper constructs IMP that is a part of pre-processing containing expandability of a file format that provides fast access to the mobile device and deals with its heterogeneity. IMP categorizes Importer modules by layer, according to map format, extracts their core elements, and reduces their weight and gives them attributes [11].

DXF/DWG Importer extracts the geographic information data property of six sections of DXF and DWG formats. BLOCKS and ENTITIES sections are used when extracting physical information and analyzing through ENTITY Parser. ENTITY Parser makes symbols or marks used in BLOCKS section into groups. Grouped information is used when indicating buildings, farmlands, roads, or rivers, and it is displayed as INSERT in the ENTITIES section. INSERT information uses group code layers defined in the BLOCKS sections, and saves them into a Temp Block Data distinguishing them from the ENTITIES section. INSERT property generates the Final Object, adding coordinates defined in the BLOCKS section onto coordinates defined in the ENTITIES section.

SHP brings shape information through the File Importer and property information through the DBF Importer. Shape information extracts the Bounding Box, which is the maximum and minimum value of coordinates of the main header, from the *.shp file, and defines the area of a shape. Shape File Handler approaches the record contents in size of the Content Length of the record header and opens the content. Record content extracts the contents according to the types of shapes at the Shape Type Importer, and waits to match the contents with property information to be extracted from *.dbf files. DBF Importer reads *.dbf files and stores its property value into the Shape Attribute Table. Shape Attribute Table matches its values with extracted shape information from Shape Type Importer and generates the Final Object. Fig. 2 shows the structure of the DXF/DWG and SHP Importer module.

Light-Weight Module categorizes the Final Object generated by DXF/DWG and SHP Importer into specific layers, thus reducing its weight – it refers to national geographic standard code when categorizing. Extracted layer code is in 4000s (4000 ~ 4637) indicating buildings, 3000s (3000 ~ 3999) indicating roads, and 9000s (9110 ~ 9233) indicating texts. Fig. 3 shows a file extracted from the DXF file of “3 Ga, Hanok Village, Pungnam dong, Wansan gu, Jeonju, Jeonbuk, 560-033 Korea” by IMP.

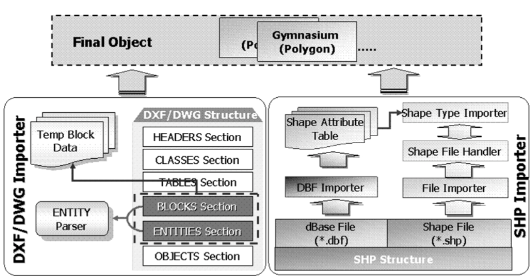

Fig. 2. DXF/DWF and SHP Importer

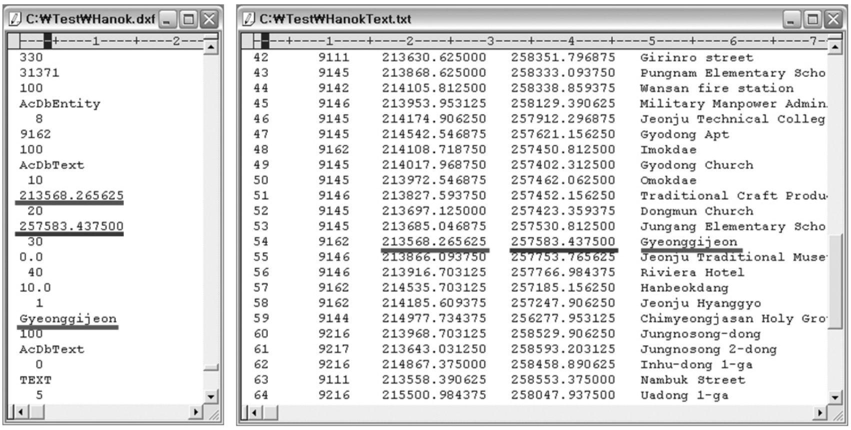

Fig. 3. Comparison between DXF file and a file extracted by IMP

3.3 Hierarchical POI

VisualGML does not provide simple map visualization but provides POI service that contains location information of individual objects. POI service constructs 3-Layer POI for efficiency of fast search, add, delete, and update of location information [12]. 3-Layer POI structure refers to the DXF standard value map (Fig. 4).

The Meta Layer defines a user-centric meta code for buildings with layer code '4' according to DXF standard code. This gives identical specific key values to the categories that fall into the same group, so that they can be distinguished from others. Meta code is divided into U-Meta and L-Meta to specialize internally its POI information. U-Meta code is defined as the upper group and is divided into 11 types of code. Each U-Meta code has L-Meta as a lower code, based on its characteristics. Key value is assigned to its names based on Name Layer. Contents Layer contains POI information, such as name, address, phone number, specifics, or webpage that are to be shown to users. POI constructs POI DB defining 14 fields (Object ID, Gu_cd/Dong_cd, Blg_nm, X_coord/Y_coord, Shape_area, add, detail-add, Tel, EX, URL, Key, O/X).

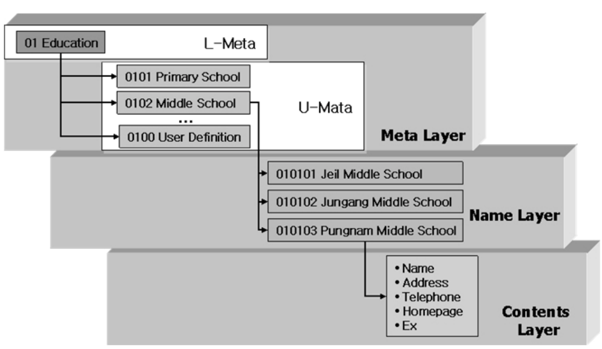

Fig. 4. 3-Layer POI Structure

3.4 GML Maker

GML Maker reconstructs geometric properties and property information extracted by IMP into the GML file. Fig. 5 shows the specific structure of GML Maker.

GML is displayed in different ways based on its map format. Attribute Converter changes the data structure of heterogeneous geographic information to parse them into common GML format, and matches properties of Final Object and GeoMetricData. GML Document Maker provides methods that generate GML schemes and property elements, referring to Attribute Define that had defined GML property tags. The generator method is composed of a generator method, which deals with basic schemes such as defining NameSpace, generating property, starting tag, ending tag, CDATA configuration, and an element generator method that deals with polygonal properties of Geometry Types, such as Polygon, Polyline, or Circle. GML Creator sorts Geometry Types by GeoMetricData, and invokes an element generator method and inserts it to the basic scheme. When insertion is complete, GML Creator brings specific field information among information in the POI DB. FeatureMember generates the GML with Exporter. Fig. 6 shows the GML file generated by adding POI information to the DXF file.

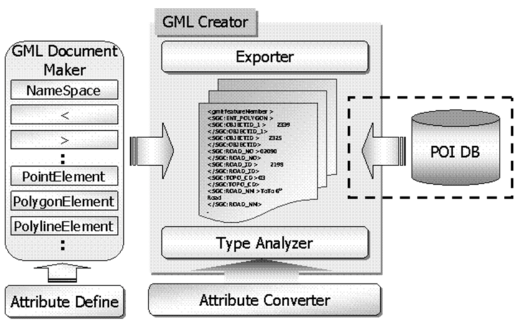

Fig. 5. GML Maker Structure

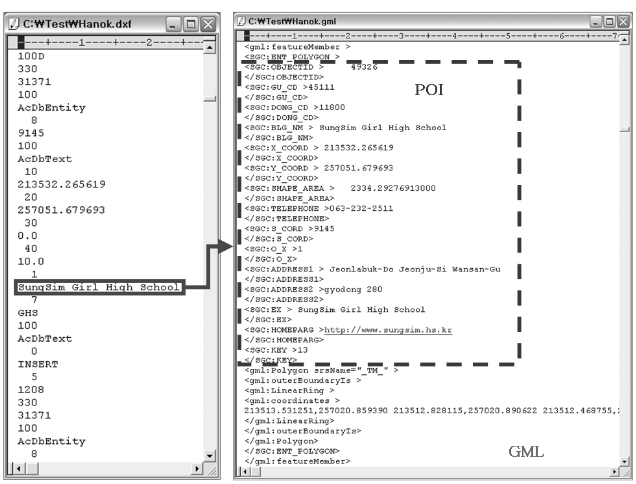

Fig. 6. GML file generated by adding POI information to the DXF file

3.5 GML Parser

GML Parser induced GML parsing through XML Parser. This enhances memory usage and CPU resource usage by the SAX method. Among the standard library modules provided by SAX, GML Parser inherits the XMLParser class and defines the GMLParser class. GMLParser class defines individual elements based on GML Schema, and invokes a handler that is defined as an event-type form to process the individual elements.

GMLParser class processes a tag with the key and the value of an element. The key is name of the tag, and the value is the name of function that will carry on starting and ending tag and is composed of a tuple. For example, if the tag is *<vgml id="2" type= "simple">*, GMLParser class invokes *vgml_start_tag[{'id':'2', 'type'='simple'}]*. Function is invoked out in the following method: in the case of starting tag, handle_starttag function is invoked, and in the case of ending tag, the handle_endtag function is redefined and altered. Fig. 7 shows the GML Parsing process.

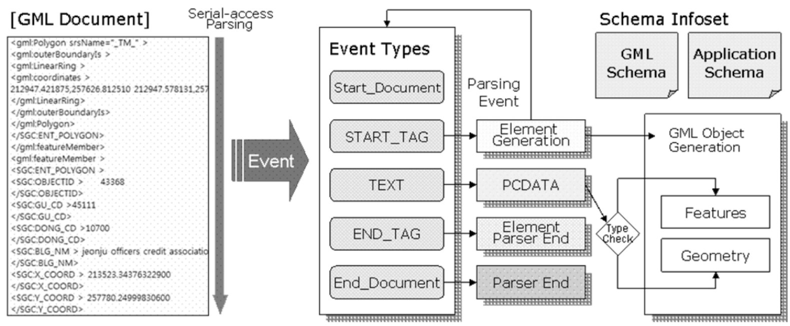

Fig. 7. GML Parsing Process

4. TRACE MONITORING VISUALIZATION OF VISUALGML

The visualization area of VisualGML is "3 Ga, Hanok Village, Pungnam dong, Wansan gu, Jeonju, Jeonbuk, 560-033 Korea". Fig. 8 shows the running window that displays map visualization with POI and tracking monitoring of the mobile device [13]. The window is composed of a main window that visualizes the map, a hierarchical POI search window, and a user event window that enables expand, minimize and move. POI visualization emphasizes buildings and their names that were searched under *Education-High School* in the hierarchical POI search tree. The selected 'sungsim girls' high school' is displayed as a pop-up of 280*100. Tracking Monitoring displays coordinates of mobile devices connected to a current server on an 800*700 size map. Movement information of three mobile devices was traced and monitored with different colors: red indicates Mobile_Device_1 moved from 'jungang elementary school' to 'cheongsoo pharmacy', green indicates Mobile_Device_2 moved from 'cheonju tradition museum' to 'sungmoon church', and blue indicates Mobile_ Device_3 moved from 'deungyongmoon computer academy' to 'myongji oriental medicine clinic'.

Fig. 9 shows that mobile devices with IDs of "1," "2," and "3" connected to a map server and visualizes GML, having moved a certain distance.

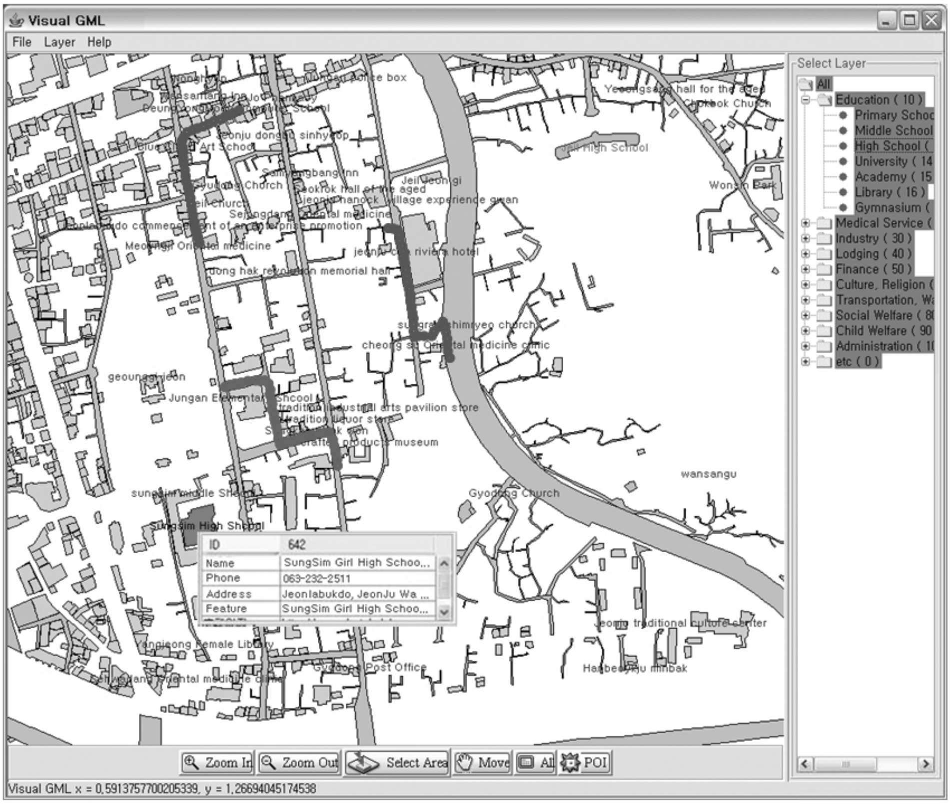

Fig. 8. VisualGML Visualization with POI and Trace Monitoring

Fig. 9. Movement Visualization of Mobile Devices

5. CONCLUSIONS AND FUTURE WORK

This paper constructed VisualGML. VisualGML generates and visualizes map information based on GML, a standard format of OGC on a mobile device. In the process of converting maps, such as DXF, DWG, SHP into GML, VisualGML embodies IMP for light-weight map information data. It developed GML Maker and GML Parser for standard map visualization. VisualGML visualized hierarchical POI to provide the GML-based map with user-centric location information. Specifically, IMP extracted unnecessary parts, whilst preserving the file's characteristic through DXF/DWG and SHP Importer, for higher efficiency of the data process capacity and data storage volume. Hierarchical POI established POI DB based on a 3-Layer POI structure, so that it can provide location information, as well as a hierarchical search method. VisualGML provided monitoring visualization that can gain and save movement information of mobile devices. The main contributions of this paper are as follows; i)visualization and generation of GML of heterogeneous geographic information data, ii)light-weighting of geographic information data, iii)user-centric visualization by hierarchical POI information, iv)real-time trace monitoring of mobile devices.

In future work, it is necessary to develop a VisualGML service that makes use of the public DXF, which is a popular geographic information service, and SVG obtaining vector graphics easily in the internet environment, to apply DEM (Digital Elevation Model) as measured topographic data to represent real space information, and to apply LOD (Level of Detail) as the rapid handling technology on all areas classifying the distant and close locations, and controlling the degree of accuracy.

We will simulate the efficiency of the data process memory and low connection speed with respect to the memory usage and the ratio of the data packet size in VisualGML respectively to consider the execution environment.

REFERENCES

[1] OpenGIS Consortium, Inc., "OpenGIS Location Service Core Services", http://www.openeospatial.org

[2] K. Virrantaus, J. Veijalainen, J. Markkula, "Developing GIS-Supported Location-Based Service", Web Information System Engineering, Proceedings of the Second International Conference on, Vol. 2, 3-6, Dec. 2001.

[3] Shashi Shekhar, Ranga Raju Vatsavai, Namita Sahay, Thomas E. Burk, Stephen Lime, "GML, Interoperability, and Standards: WMS and GML based interoperable web mapping system", Proceedings of the 9th ACM international symposium on Advances in geographic information systems, Nov. 2001.

[4] Autodesk Drawing eXchange Format, http://www.autodesk.com/techpubs/autocad/acadr14/dxf

[5] ESRI, ESRI Shapefile Technical Description. ESRI, INC, http://www.esri.com, 1998.

[6] TatukGIS Inc. TatukGIS Viewer 1.4, http://www.tatukgis.com/products/Viewer/viewer.aspx

[7] Safe Software Inc. FME Universal Viewer 2003, http://www.safe.com/products/fme

[8] Snowflake Software Ltd. OS Master Map Viewer 2.0, http://www.snowflakesoftware.co.uk

[9] eSpatial Inc. iSMART Explorer4.4, http:/www.espatial.com1.

[10] CyberMapWorld co., Location Tracing Solution, http://lbs.cybermap.co.kr/

[11] C.Y. Jeon, J. Park, E.H. Song, Y.S. Jeong, "A Development of Integrated Map Preprocessor for Mobile GIS Visualization, " KIPS, Vol.12 No.1, pp. 707-710, 2005.

[12] Eun-Ha Song, Laurence T. Yang, Young-Sik Jeong, "Visualization of GML Map using 3-Layer POI on Mobile Device," International Conference of Embedded Software and Systems(ICESS 2007), LNCS 4523, 14-16 May. 2007, pp.328-337.

[13] Eun-Ha Song, Sung-Kook Han, Laurence T. Yang, Minyi Guo, Young-Sik Jeong, "A GML-based Mobile Device Trace Monitoring System," Embedded and Ubiquitous Computing Workshops 2006 (EUC Workshops 2006), LNCS 4097, Aug. 1-4. 2006, pp.234-243.

Eun-Ha Song

She received a B.S. degree in statistics and an M.S., and Ph.D. degree in computer engineering from Wonkwang University, Iksan, Korea in 1997, 2000, and 2006, respectively. Since 2007, she has been with the Department of Computer Engineering, Wonkwang University, Iksan, Korea, where she is a full-time lecturer, working in the areas of Grid Computing and Cloud Computing, Distributed and Mobile computing, and USN middleware.

Young-Sik Jeong

He received a B.S. degree in mathematics and an M.S., and Ph.D. degree in computer science and engineering from Korea University, Seoul, Korea in 1987, 1989, and 1993, respectively. Since 1993, he has been with the Department of Computer Engineering, Wonkwang University, Iksan, Korea, where he is a professor, working in the areas of Grid Computing and Cloud Computing, Distributed and Mobile computing, and USN middleware.