Mobile Object and Real Time Information System Modeling for Urban Environment

Adil El Bouziri[*1], Azedine Boulmakoul[1], Robert Laurini[2]

[1] LIST Lab., Computer Sciences Department
Mohammedia Faculty of Sciences and Technology (FSTM),
B.P. 146 Mohammedia, Morocco.
{a_elbouziri@yahoo.fr, boulmakoul@menara.ma}
[2] LIRIS- INSA de Lyon, 20 Avenue Albert Einstein, F-69621 Villeurbanne cedex France.
{Robert.Laurini@insa-lyon.fr}

Abstract. *With the convergence of wireless communication, mobile positioning and GIS technologies, the demand for location based services (LBS) has already appeared and prompted research in different fields. Location based applications and services deliver geo-data and real-time information to mobile users in accordance with their locations and preferences. In this paper, we present our data model of mobile object on multimodal transportation network, which extends Open GIS Simple Feature specification. This model intends to improve the development of mobile applications and the location based services that deal with tracking and management of mobile objects. Then, we propose a real-time object oriented model for location based services with the RT-UML specification that adds standard real-time extensions to UML. The component based modeling with UML 2 is also introduced. A prototype system based on this architecture has been developed as mobile tourism guide system for Mohammedia City.*

Keywords: Mobile object, Mobile applications, LBS, RT-UML, object oriented model, J2ME, Web services

1. Introduction

For several years, new geographical and urban applications have been emerging in which characteristics of communications and real time have been very important. We are facing a new discipline called telegeomonitoring which can be considered as a discipline characterized by positioning systems, cartography, the exchange of information between different sites and real time spatial decision making [9]. Telegeomonitoring system development combines two heterogeneous technologies: the geographical information systems (GIS) and telecommunications technology [5].

In addition, during the last years wireless communications have experienced a spectacular growth. Most of the population is already familiarized with the use of devices like mobile phones, Personal Digital Assistants (PDAs), etc. This factor and other point out the great business opportunity which is those services that can be used by a great mass of customers through a mobile phone. Location Based Services (LBS), that can be classified under the umbrella of telegeomonitoring, refers to the wireless services provided to the subscriber based on his current location. The position can be known by getting it from mobile phone network, or from another positioning service, such as global positioning system (GPS). The location-based system is complex and requires the seamless integration of many different technology components into one system. The core technology for any LBS solutions is GIS system, which performs important functions such as determining street addresses, look up landmarks, calculate optimal routes and render custom map. Moreover, the integration of geospatial information and mobile computing is driven by market demands and technologies. This system creates a new channel of business practice, and thousands of potential applications and services can be developed. Hence, it is exploring a new era of mobile geographic information services [4][8].

Some examples of mobile applications and LBS are listed below [6]:

---

[*] Corresponding author

- Emergency services that could be drive oriented according to the user spatial position.
- Service Information, such as tourist information ("what are the hotels near me?")
- Objects tracking
- Traffic information

However, the modeling and the management of a mobile object are fundamental in this type of applications to furnish the requested services. It is necessary to provide a mobile object data model for representing and querying mobile objects especially those with point geometry moving on a transport network. In this paper, we present our mobile data model on multimodal transportation network to improve the development of mobile applications and deploy it in a proposed real-time object oriented model for location based services. We introduce also the component based modeling with UML 2 [18], which permits to reuse sub-systems components in many other contexts.

The RT-UML specification is the UML profile for schedulability, performance and time [19]. It has recently appeared to improve the use of the object orientation in the development of real-time applications. It used to model this system in order to represent real-time tasks with UML and to facilitate development with real-time modeling notation.

The rest of the paper is organized as follows: Section 2 presents an overview of RT-UML specification. Section 3 describes a generic topological class diagram of a multimodal transportation network. In section 4, we propose a data model of mobile object on transportation network. Section 5 gives an object oriented model with the RT-UML specification of the proposed architecture for location based services, and introduces component based modeling. In section 6, a prototype system concerning mobile tourism guide system for Mohammedia city is described.

2. RT-UML specification

The RT-UML specification is the UML profile for real time modelling, formally called schedulability, performance and time (UML-SPT) [19]. It has been adopted by the Object Management Group (OMG). This increased the interest in the use of the object oriented technology and UML, in particular to model and build real time systems. This profile is designed to add standard real-time extensions to UML in order to facilitate development with real-time modeling notation. This profile allows the construction of predictable UML models and it focuses notably on key system properties such as timeliness, performance, and schedulability. RT-UML is a framework to model quality of service, resource, time and concurrency concepts. Actually, it provides the user (modeler) with a set of stereotypes and tagged values in order to annotate the UML model. Quantitative analysis (schedulability and performance analysis) can then be applied to these (predictive) UML models.

The structure of the profile is modularized to allow users to only choose the elements that they need. The profile defines a basic framework which is a set of sub-profiles that represent the general resource modeling framework. This framework is divided into three packages:

- Resource modeling for the basic concepts of quality of service and resource
- Concurrency modeling for concurrency modeling
- Time modeling for time and time-related mechanisms modeling

Derived from the basic framework, there is a sub-profile used for schedulability analysis of systems. This model is more interesting for real-time systems where the question of when a response to an event occurs is very important for the correct behavior of the system. The schedulability sub-profile focuses on how to annotate the model in ways that allow a wide variety of schedulability techniques to be applied. Figure 1 depicts the metamodel defined in RT-UML for the main concepts involved in the schedulability analysis: the execution engine, threads (task or process), shared resources, external events and the response of the system to the external events. To representing these concepts in UML, a set of
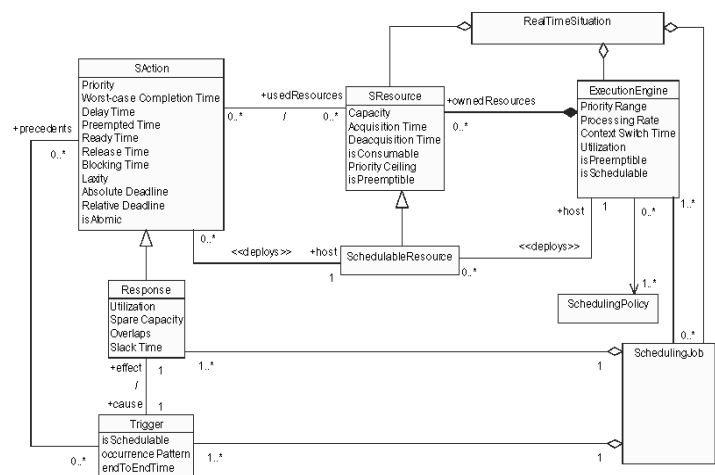

Fig. 1. Core schedulability model From [19]

stereotypes and their associated tagged values is defined in schedulability sub-profile. The table 1 presents a sample of stereotypes. The application of these stereotypes will be illustrated in our real time object oriented model in section 5. The ability to undertake quantitative analysis at early phases of the development process is important to reduce the cost.

Stereotype	Real-time Concept	UML Model Element
«SASituation»	Real-time situation	Collaboration, Sequence diagrams
«SATrigger»	Event	Message, Stimulus
«SAResponse»	Response	Method, Action
«SAAction»	Action	Method, Stimulus, Action
«SASchedulable»	Task, Thread	Instance, Object, Node
«SAResource»	Resource	Instance, Class, Node
«SAEngine»	CPU, Processor	Object, Class, Node

Table 1. SPT common stereotype for schedulability analysis

3. Multimodal transportation network model

The multimodal transportation network is modeled as an oriented graph, whose fundamental elements are nodes and links. For topological aspect, we prefer to use the concept of a node instead of a point. We shall start our spatial network model by focusing very sharply on the definitions and semantics of these two entity types and the relationships between them. A *node* (for generic topology) is the smallest identified location in space. It can play many different roles in the transportation network (node is not just a location in space). A *link* is the unique oriented path which connects two nodes. We can also introduce hypernodes and hyperlinks entities in this modal. A *hypernode* is a node composed of one or more nodes, i.e., a node is a station for a single transportation mode and an hypernode is an intermodal station, that is a place where people can enter or leave the transportation network or change their mode of transport. A link is an unidirectional path. A *hyperlink* is a link connecting two hypernodes, and it is composed of one or more links [4][5].

Figure 2 gives a logical view of spatial networks. The node, the link and the relationships between them are considered as a generic structural pattern which specifies many concrete structures in a transport network.

4. Data model of mobile object

In this section, we present a data model of mobile objects moving on the transportation network. The model is represented by spatio-temporal classes with mobility aspects. The location based services are concerned by the mobile point objects, i.e. objects with zero extent that change their location continuously over a predefined network infrastructure. Thus, our emphasis is put on the modeling the mobile point object and its relationships with the main classes which represent the multimodal transportation network.
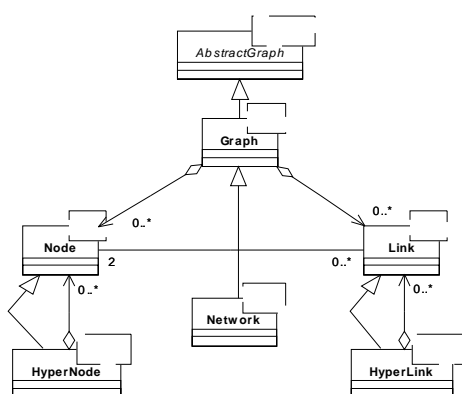

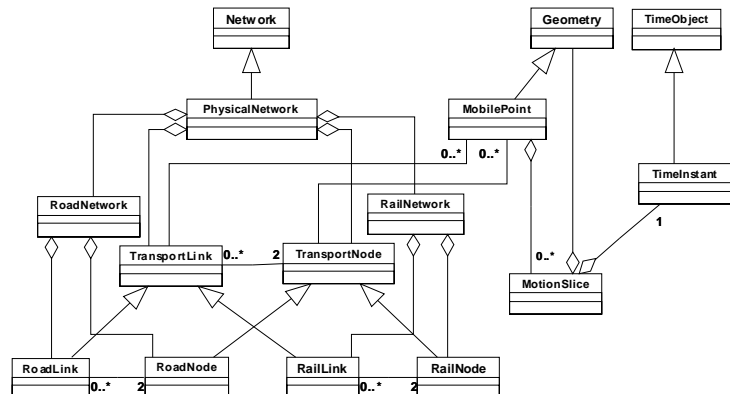Figure 2. Class diagram of multimodal transportation network

Figure 3. Data model of mobile object

To represent transportation network, a *network* class is added. It inherits *graph* class. The *PhysicalNetwork* class, which is a specialization of a network class, is comprised of a road network and a rail network, in which the transport services are supposed to run (see Figure 3). The basic entities of the physical network are *TransportLink* and *TransportNode.* The road network represents all carriageways available for vehicles (cars, buses, etc.) and into which the mode lines can be inserted. Two entities: *RoadLink* (carriageway available for cars, buses …) and *RoadNode* (connection between road segments) are basic elements of the road network. In similarity, the description of the rail network is meant to be a model of the track network along which vehicles (or trains) can physically proceed. It is modeled by two entities *RailLink* (track along which metro or train can physically proceed) and *RailNode* (located at switches).

On other hand, the data model of the mobile object appropriately extends the OGC Simple Features model, which defines abstract *Geometry* class, and its hierarchy of the specialized geometric classes (*Point*, *LineString*, *Polygon*, *MultiPoint*, etc.) [22]. The time dimension of a mobile object is specified through the *TimeObject* class hierarchy, defined in accordance with ISO TC 211 Temporal Schema (*TimeInstant*, *TimePeriod*, *TimeDuration*, etc.) [23]. The *MobilePoint* class provides modeling mobile point objects that move continuously over a predefined network infrastructure. Since it inherits the OGC Geometry class, this class and its specialized classes can be treated in the same way as any other geometric object. The *MobilePoint* defines predicates and operations for the management and querying mobile objects with the respect to the OGC and ISO 211 specifications. Our approach is based on the comprehensive framework of the data types, the rich algebra defined in [7] and the works related to location data models and query languages[13-15]. In addition, the model describes relationships between *MobilePoint* class and the main classes of the transportation network, such as *TransportLink* that is the way where the mobile object can move. The *MotionSlice* class provides the representation of the complete motion of the mobile object. An instance of this class, aggregated by the *MobilePoint* class represents the registered location of the mobile geometry.

5. Overall system architecture

5.1. Architecture

The architecture of the proposed system for location based services is depicted in Figure 4. It is complex and requires the seamless integration of many disparate technologies in one system. Mobile applications are a good way in this system to present real-time information that has a great impact on the customer's satisfaction. A global positioning system (GPS) receiver will be needed in order to determine the current position of the mobile user and also to send periodic updates of the mobile user's position to the location server.

Figure 4 shows the main components of the considered architecture:

a) **Mobile Object (MO)**: represents moving object like vehicles or travelers with a location detection mechanism such as GPS. This entity can be modeled with the proposed data model previously presented. It can be represented by *MobilePoint* class defined in our mobile object data model.
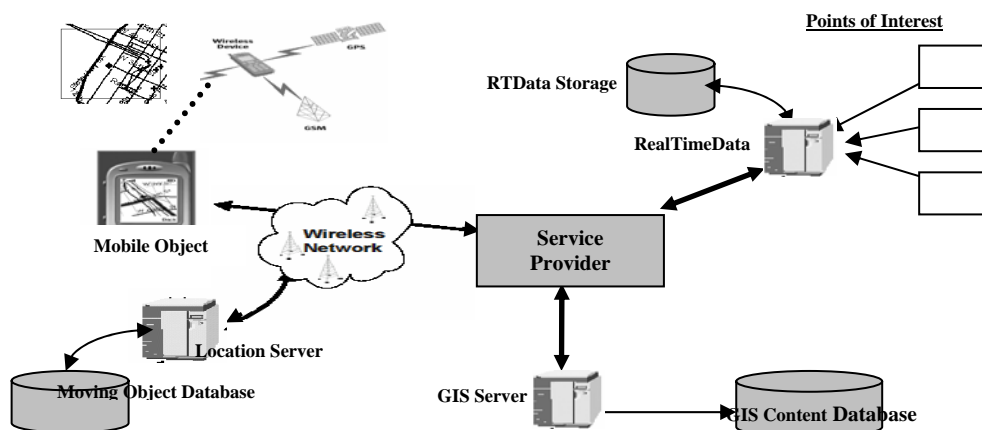

Figure 4. Overall system architecture

b) **Service Provider (SP)**: coordinates between different components of this system to provide anywhere, anytime real-time data, map or other services to the mobile object related to its spatial position. It can use web services [20] that permit to interact with any type of mobile devices to furnish the desired services. The web services can communicate and coordinate with other web services developed with different technologies.

c) **Location Server**: in order to determinate any mobile object positions, the database servers storing the user's location will be generally distributed among cellular architecture network [10]. We can use in our system a simple architecture in which a Location Server is connected to whole local database servers related to a number of cells. These moving object databases have to deal with the moving objects and all kind of spatio-temporal queries [7]. The location server (Index Server) holds for every object its local database server.

d) **GIS Server**: it has a suite of tools to perform spatial operations which include geocoding, reverse geocoding, routing and several other services. It needs to access GIS Content database to perform its functions. It can also access via Internet network other GIS servers and GIS databases.

e) **RealTimeData**: represents an entity that is accessed concurrently and receives real-time data from different data sources like congestion sensors or points of interest: local restaurants advertising their menus, hotels announcing room availability, theatres listing last-minute tickets for sale, availability of parking, etc. It analyses up to date information and stores the processed data in RTData Storage database. The Service Provider may interact with this server to retrieve real-time data.

5.2. Real time object oriented model

In this section, we propose an object oriented model with real-time extensions following, as possible as, the definition found in the open GIS specification [21] and using the recent specification RT-UML [19]. Figure 5 shows a collaboration diagram that emphasizes the structural organization of the objects that exchange messages.

This system permits mobile users to access information related to location, such as ("where are the hotels and parking near here (my current position)?").

In addition, the mobile user can get real-time information like parking availability, hotel information about promotion and discount tariffs or the closest empty taxis.

The following steps describe the interaction between components in the system. These sequences of events take place to provide requested service and improve mobile application:

- The Mobile Object, represented by the *MobilePoint* class, requests via wireless network real-time desired services from the Server Provider.
- After identification of the subscriber and the requested service, the SP parses, evaluates and interprets this spatio-temporal request and call different components in order to provide mobile geographic services to mobile client. The SP formats the request data to Location Server to have positions of all mobile objects that are mentioned in the MO request.
- The Location Server validates Service Provider's identity and request format. Then, it retrieves the relevant positioning data from the moving object databases. It constructs a message which consists of the positioning data and other supporting elements such as GMT and local time. It sends it to the SP.
- The SP parses this message to get positioning data. Then, it opens a connection to GIS Server to send a map request or request searching some spatial objects "such as parking" in whose influence areas the MO is found.
- The GIS Server sends its response to the SP.
- Since the SP has now information about the MO, mobile objects and spatial objects, it can interact with RealTimeData entity to retrieve real-time information concerning them.
- Finally, the SP sends its response to the Mobile Object describing the service.
- Mobile device application installed on mobile terminal permits to parsing the response receiving from the SP. It allows subscriber to view the processed map with services and plotted position, and interact with other functions.

The collaboration diagram is annotated with several stereotypes from the schedulability sub-profile. For example, *PointsOfInterest* class is associated with « *SAschedRes* » stereotype. Instances of the classes that are associated with this stereotype execute concurrently in the application context. The execution flow of « *SAschedRes* » stereotype is identified as a scenario which is started after an activation message stereotyped with « *SAtrigger* ». During this execution, many actions stereotyped by « *SAaction* » with a specified priority (*SAaction.SApriority*) may be executed in response, for instance, to a method call. The basic structure of the class scenario is characterized by the « *SAresponse* » stereotype and executes periodically

after a periodic event of the trigger associated to the tag *RTat=('periodic',50,'ms')*. In addition, the *RealTimeData* class stereotyped with « *SAresource* » is a protected resource that is accessed concurrently using mutual exclusion mechanisms. The *SAcapacity=1* means that one element can simultaneously access to an instance of this class. The *"SAaccessControl"* tag defines the access control policy.

   Concerning transition to implementation, we may use *Real-Time Specification for Java* (RTSJ) for generating code from the collaboration diagram, once the RTSJ offers concepts that are logically related to the RT-UML timing constraints [2][3].
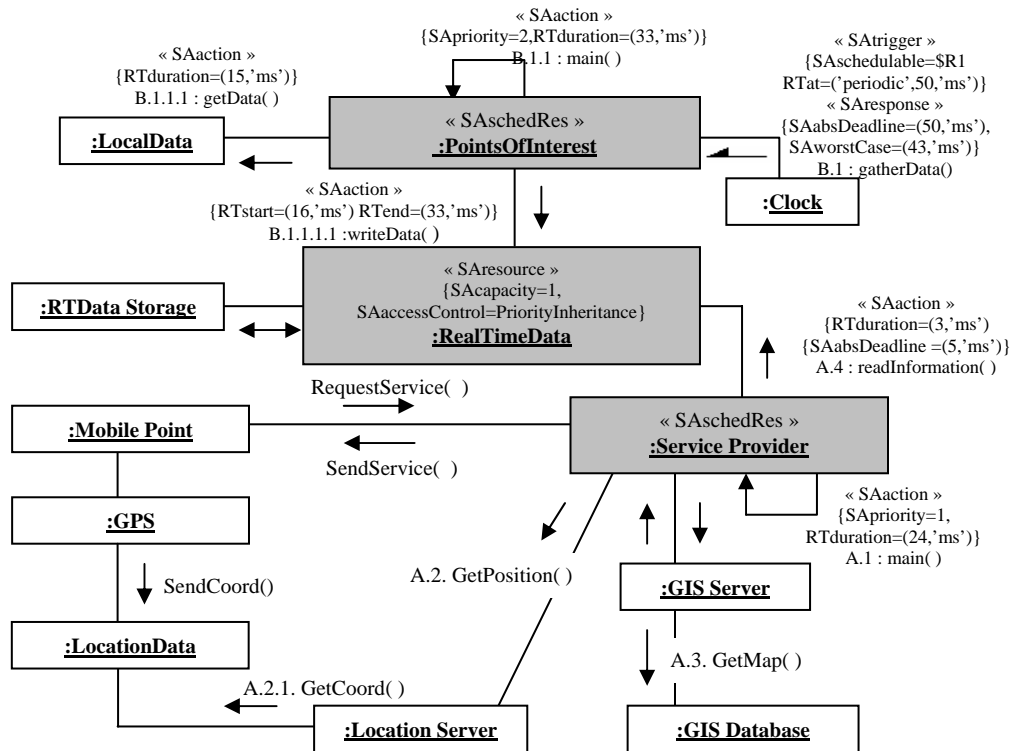

Figure 5. Collaboration diagram

5.3. Modeling with components

We will use the concepts of components in UML 2 [1][11][18]. With regard to UML 1.x, this concept has been modified by addressing now system structures. It is from the main improvements in UML2 which supports the component based development via composite structures.

A component is a modular unit with well-defined interfaces. The interfaces of a component are classified as provided interfaces and required interfaces. Provided interfaces have defined a formal contract of services that the component provides to other components while required interfaces have defined the services that it requires from other components in its environment to operate properly.

In UML 2, a component can have two different views, external view and internal view: The external view is also known as a "black-box" view in which it exhibits only the publicly visible properties and operations which are encapsulated in the provided and required interfaces. The wiring between components is specified by dependencies or connectors between component interfaces. The internal view is a sort of "white-box" view which shows the component internals that realize the functionality of the component. An external view is mapped to an internal view by using dependencies which are usually shown on structure diagrams, or by using delegation connectors that connect to the internal parts which are shown on composite structure diagrams.

The structure diagram in figure 6 shows the composite structure of components. The wiring between components is represented by assembly connectors between provided and required interfaces. This component-based structure aims to hierarchically decompose the complex system into smaller sub-systems and then connect these sub-systems together. We can reuse any part of the modeling system in many other context. In this component based modeling, based on the data model of mobile object, we make up two components relative to mobile object: a module in the server and another in the client. The component of the

client has some additional classifiers (classes or components) and interfaces with purpose to deal with the location capture and to calculate the uncertainty. These issues are discussed in [15]. In addition, the FSNet-solver component represents a fuzzy spatial network solver component based on a new algebraic structure proposed in [4][5] to solve a path-finding problem in a fuzzy graph. This component permits to solve the problem of the k-best fuzzy shortest paths [4].

Focusing the Service Provider in figure 6, we see its external view in the middle of the figure. The internal view of design for this component is depicted in figure 7 that shows how some Service Provider's parts that are connected to each other.
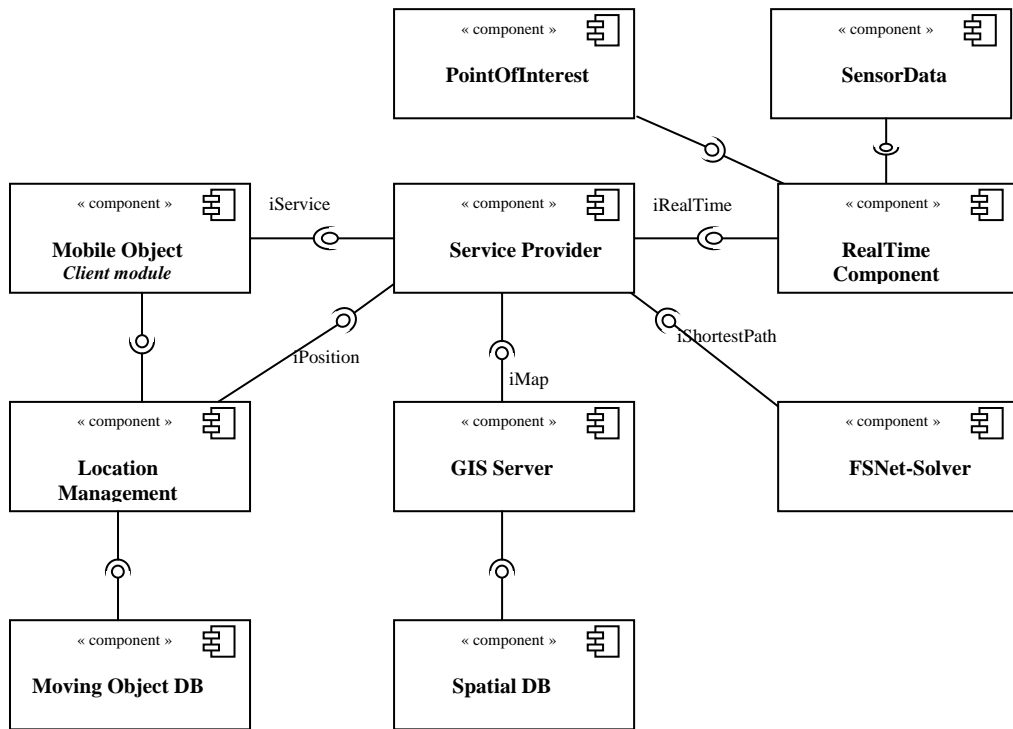

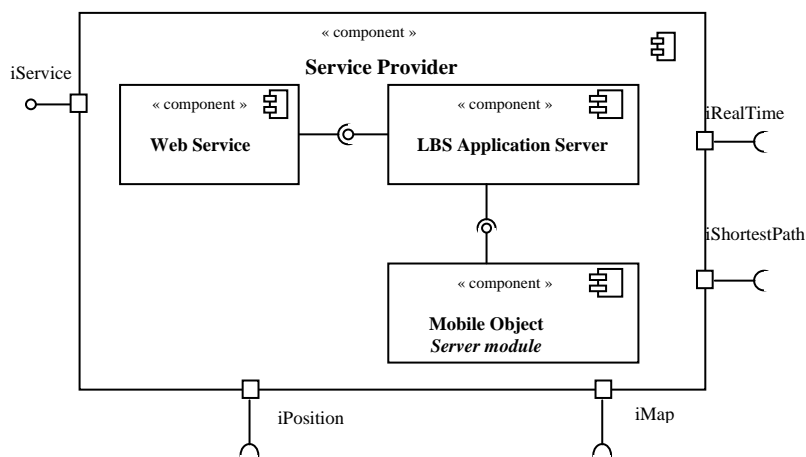Figure 6. Composite structure of overall system

Figure 7. Internal view of Service Provider component

6. Prototype system

6.1. Server side implementation

Based on our architecture, a prototype system as LBS example has been developed. It concerns Mohammedia tourism guide project. The server side employs Java SDK version 1.5 and MapObjects Java Edition from ESRI® (version 2.2) as a mapping tool. The MapObjects software is employed to provide a range of GIS functions (e.g. spatial and attribute querying, geocoding, etc…) and represent diverse computational results in series of color maps with proper scales and legends. This architecture uses as application server Sun java system application server (edition 8.2) and PostgreSQL version 8.1.3 as a GIS database server.

Furthermore, to furnish requested services to mobile user and provide up to date information concerning some landmarks, a web service has been implemented. It offers requested information to mobile user like position of the nearby landmarks or real-time information concerning a particular landmark. We have used Netbeans IDE 5.0 that comes bundled with web service support based on JSR-109, which is a development paradigm that is suited for J2EE development, based on JAX-RPC (JSR-101) [12]. To consuming this web service, we have to retrieve the WSDL file that describes the web service's interface.

6.2. Creating a Java ME web services (JSR172) client

For mobile client, a mobile application has been developed. It comprises a midlet that can connect to web service to get a desired service. Mobile device or emulator platform must support the JSR-172 specification "Java specification request" [16]. The JSR 172 specification is the J2ME web services specification that defines a standard set of APIs for XML processing, and SOAP web service clients on the J2ME platform.The Netbeans 5.0 mobility pack is used with wireless toolkit version 2.3 beta which supports JSR-172 and JSR-179 specifications [16][17]. JSR-179 defines an optional package that will enable developers of mobile applications and LBS to improve the development of mobile location based applications for mobile devices.

In this application, the simulation of mobile object position is generated from a file that contains latitude, longitude and altitude information. Figure 8 demonstrates the mobile tourism guide for Mohammedia city on an emulator of mobile device. It illustrates how to construct a location based application using JSR 172 and JSR 179 API. The mobile user can ask for example the nearby landmark like Parking, restaurants or hotels. Then the Mohammedia map is generated with a moving icon that represents the mobile client position. Other icons show the requested landmarks that are near to mobile client. The latter can also receive real-time information concerning the desired landmark such as the availability of the closest parking.

Figure 8. Screenshots of mobile client application

7. Conclusion

The main contribution of this paper is the data model of mobile object and its main relationships with the transportation network. This model will strongly improve the development of location based services and mobile applications. An object oriented model with the emerging RT-UML specification for location based services is proposed. In addition, we introduce UML 2 component based modeling of the proposed architecture. The prototype system has been developed and is successfully worked with an emulator of mobile devices. This project concerns the mobile tourism guide system for Mohammedia city. The J2ME device application is designed with JSR 172 and JSR 179 specifications to test the whole system and identify appropriate communication mechanism for delivering location data and real-time information to the new generation of the smart phones. Furthermore, we will also explore more security issues, barriers and opportunities to the deployment of location based services and applications.

References

[1] M. Björkander and C. Kobryn, "Architecting Systems with UML 2.0," *IEEE Software*, July 2003, p57-61

[2] G. Bollella, J. Gosling, and al., *The Real-Time Specification for Java*. Addison-Wesley, 2000.

[3] A. Boulmakoul and A. E. Bouziri, "Applying RT-UML and RTSJ specifications for mobile Tourist Information System," *SITA'04*, Rabat, Morocco, December 2004.

[4] A. Boulmakoul, A. Laurini, H. Mouncif and G. Taqafi, "Path-finding operators for fuzzy multimodal spatial networks and their integration in mobile GIS," *In Proceedings of the 2th IEEE International Symposium on Signal Processing and Information Technology*, Marrakech, Morocco, December 2002.

[5] A. Boulmakoul, "Fuzzy graphs modelling for HazMat telegeomonitoring', *European journal of operational research*, (175) 2006, 1514-1525

[6] O. L. M. de Farias, J. A. Ribeiro and C. P. de Oliveira, "An object oriented model for dynamic geographic information systems and location based services," *In Proceedings of 6th AGILE Conference on Geographic Information Science*, Lyon, April 2003.

[7] R. H. Güting, and al., "A foundation for representing and querying moving objects," *Geoinformatica*, ACM Transactions on Databases Systems, Vol. 25, No. 1, 2000, p1-42.

[8] A. Jagoe, *Mobile Location Services- The Definitive Guide*, Prentice Hall PTR, 2002

[9] R. Laurini, "An introduction to TeleGeoMonitoring: Problems and Potentialities," *In Proceedings of GISRUK '99*, UK, April 1999

[10] E. Pitoura and G. Samaras, "Locating objects in mobile computing," *Knowledge and Data Engineering*, Vol 13, no.4, 2001, p571-592.

[11] D. Servat, H. Dubois and S. Gérard, "le concept de composant dans UML2," Rapport interne IRCCyN RI2005_1, CNRS, Mars 2005

[12] I. Singh, and al., *Designing Web Services with The J2EE 1.4 Platform JAX-RPC, SOAP, and XML Technologies*, Addison Wesley, 2004.

[13] D. Stojanovic and S. Djordjevic-Karan, "Modeling and querying mobile objects in location based services," *Facta Universitatis Journal*, Series Mathematics and Informatics, NIS, Yugoslavia. 2003

[14] M. Vazirgiannis and O. Wolfson, "A spatio-temporal model and language for moving objects on road networks," *In Proceedings of 7th SSTD*, USA, 2001, p20-35.

[15] O.Wolfson, S.Chamberlain and al., "Cost and Imprecision in Modeling the Position of Moving Objects." *In Proceedings of the Fourteenth International Conference on Data Engineering (ICDE)*, 1998

[16] JSR 172: J2ME Web Services Specification, Final Release, 2004

[17] JSR 179: Location API for J2ME Specification, Final Release 2, 2006

[18] OMG, Unified Modeling Language: Superstructure, formal/05-07-04, version 2.0, 2005

[19] OMG, "UML Profile for Schedulability, Performance and Time", formal/05-01-02, version 1.1, 2005

[20] Open GIS Consortium, "Open GIS Web Services Architecture" (WSA), OGC 03-025, version 0.3, 2003

[21] Open GIS Consortium, "OpenGIS Location Services (OpenLS^{TM}): Part 1-5 Core Services", OGC 03-006r1, 2003.

[22] Open GIS Consortium, "OpenGIS Simple Feature Specification for SQL", document 99-049, May 1999.

[23] ISO/ TC211 Geographic Information / Geomatics: ISO 19108- Temporal Schema. 2000.