Logistics Service Development based on the Geo2Tag and Smart-M3 Platforms

Kirill Krinkin[1,2], Kirill Yudenok[2], Mark Zaslavsky[1,3]
[1]FRUCT LLC
[2]Saint-Petersburg State Electrotechnical University «LETI»
[3]ITMO University
Saint-Petersburg, Russia
{kirill.krinkin, mark.zaslavsky}@fruct.org, kirill.yudenok@gmail.com

***Abstract*—In this article we will talk about implementation of a logistics service for a car fleet management based on LBS Geo2Tag and Smart Space Smart-M3 platforms. Each platform has its own role in the work of logistics service. All car fleet location information is stored and processed by the Geo2Tag platform. A Smart-M3 platform performs service metadata analysis in the form of service ontology and knowledge reasoning of these data. This work is focused on analysis and development of a logistics car fleet management service using Geo2Tag and Smart-M3 platforms. Currently, there are a different classes of geographic data processing applications of road traffic objects from In-Car LBS system, fleet management systems to a satellite monitoring system for vehicles of the country. The current solution will create a context-aware system for processing and reasoning smart space subjects geographic data using LBS Geo2Tag platform. The system will consist of a variety types of agents or services for a fleet management from various angles – manager, driver and client.**

I. INTRODUCTION

The idea of integrating LBS-systems [1] and Smart Space technologies [2], [3] is not new [4], their combination opens up new possibilities to create services in different environments (spaces). So far, smart spaces and geo-tagging systems are being developed mostly separately, there are only few works [5], [6], [7] where software design of smart spaces and geo-tagging integration is discussed.

The integration of these systems will give eventually combined capabilities of both type of systems, in conjunction with (a combination) and complementing each other. The combined capabilities will allow to create a more functional solution for your current tasks and choose the best solution for its implementation in the form of an agent, service or other use case. LBS-systems possibility to operate with geographical characteristics of real or virtual world objects will allow to use these systems for the smart space subjects for further processing and reasoning new knowledge.

This paper discusses the requirements and architectural solutions for logistics service based on Geo2Tag [8] and Smart-M3 [9], [10], [11] platforms. The Geo2Tag and Smart-M3 platforms work will be organized through a special Geo-Coded Smart Space integration agent (GCSS) [4]. As a co-option usage of the Smart Space and LBS concepts is regarded a car fleet management scenario.

The main objective of the project is to develop a common service for car fleet management with the ability to provide additional services and data associated not only with the car choice, but also to provide an actual traffic information.

Most of the current car fleet management solutions works with the data provided by service users or its employees. Analysis and processing of all context-aware information about the car fleet objects will provide an opportunity to evaluate the work of logistics service from different sides (extensive statistical analysis of all car fleet objects and the provision of additional services).

The initial logistics service functions will be the car fleet configuration and management, the ability to plan trips on a selected car, as well as collection and analysis of trips, drivers, passengers and cars metadata. The logistic service will be developed based on service-oriented architecture with the application programming interfaces (API) to the Geo2Tag and Smart-M3 platforms.

The paper is structured as follows: second chapter analyzes existing works; third chapter describes of the logistics service use-case scenario; fourth chapter discuss of a new implementation of integration LBS and Smart Space platforms, its software, components and ontological models and as well as the logistics service functionality; in the conclusion discusses the steps for current and further research.

II. RELATED WORKS

There are a large number of fleet management systems [12], [13], [14] for the current moment. Most of these systems have a mobile or web clients to work with the fleet.

Service like Uber [15] provides a user information about all available cars in his city with the possibility of rapid order and up to date information provision about it. The service is based on the model of transfer all communication channels with current and potential customers into the mobile application, which is the only interface between the customer and the business. This model allows you to seriously reduce costs, and hence the cost of services, improve service quality and offer customers new convenient services, as mobile application allows you to quickly identify the client to determine its location, and most importantly, automate routine processes of filing and payment services. In the Uber service you have access up to five kinds of vehicles, from classic taxi to the high-rated cars. Like the Uber there are similar carfleet services, for example, Yandex.Taxi [16], Lyft [17], Gett [18] and others.

As for In-Car LBS system, the most frequently used ones (Yandex.Transport, GM OnStart, Ford Sync, Chrysler USonnect, TomTom. Garmin, Google Maps Transit) provide relevant information to the driver about the road state (routes, traffic, cameras, drivers comments) [19-25]. Also, some of the such systems have functionality of the car cabin temperature control, speech commands recognition [26] and etc.

In [27] discussed the concept of the "*Connected car*", that refers to the use of in-car telematics, a range of technologies that leverage connectivity, whether over the Internet or via dedicated short-range communications (DSRC), with diagnostic, location, or other information to provide new safety, convenience, and communications services. Also in the paper discussed the possibilities offered by these systems, their processes data, privacy and security of collected data by the car onboard computer.

The "*ERA-Glonass*" project is a trucks monitoring system on all roads of the city or country based on a satellite navigation system Glonass [28]. Analogs of "*ERA-Glonass*" are European project "*eCall*" [29] and USA project "*NG-9-1-1*" [30]. *ERA-Glonass* and *eCall* projects are the one agreed protocols and system standards based on the relevant technical standards ETSI [31] and 3GPP [32]. *ERA-Glonass* car terminals as desired car owners can be used to provide a whole range of additional services related to navigation, communication, remote vehicle diagnostics, etc.

Another possible scenario for using Smart Space and LSB platforms integration is an assistance with disaster and emergency situations, for example, UAV drones creates a smart space network in the certain area of the disaster, which is used for monitoring and searching victims. It is also developing in many countries a scenario of «*Network-Centric warface*» [33] is not possible without a creation such space on a battlefield.

III. TRAFFIC MONITORING SCENARIO

A. Logistics service

Logistics service is a set of mobile clients and applications for the car fleet creation and management from all its sides. The car fleet owner can create its own fleet, engage in its management and improvement using statistics from car fleet drivers and clients. The car fleet driver performs the customers orders and transfer of all statistics about his orders and cars. The customer can use any car fleet service to accomplish its own objectives, as well as help services to evolve in the right direction.

Logistic service will combine the capabilities of two Geo2Tag and Smart-M3 platforms, which will greatly improve the efficiency of management and operations of the car fleet by collecting and analyzing data about the trips, car characteristics, client reviews and common fleet statistics. The ability to reasoning new knowledge in the work of service will reveal the hidden factors that influence to the car wear, the profits value and clients loyalty without using third-party tools.

In the previous works [4, 34] discussed the construction of a Smart Space Geo-Coded System (GCSS), which provides the following use cases:

- geographical markup of smart space data;
- search set reduction (filtration by data context);
- search context rectification.

As a result currently supports the first use case, the second one is partially implemented and requires modification, the third case is not implemented. Logistics service requires the implementation of all GCSS agent use-cases.

B. Logistic service conceptual model

Figure 1 presents a three-level conceptual model of traffic monitoring scenarios, at the example of logistics service for car fleet management. The conceptual model is divided into three levels – 1) the World level reflects the real service world, 2) the Smart World level is a real world with elements of "smart" things instead of the real objects, 3) the Devices and Services level enhances the Smart World level with the help of devices and services and links between these objects in the Smart Space.

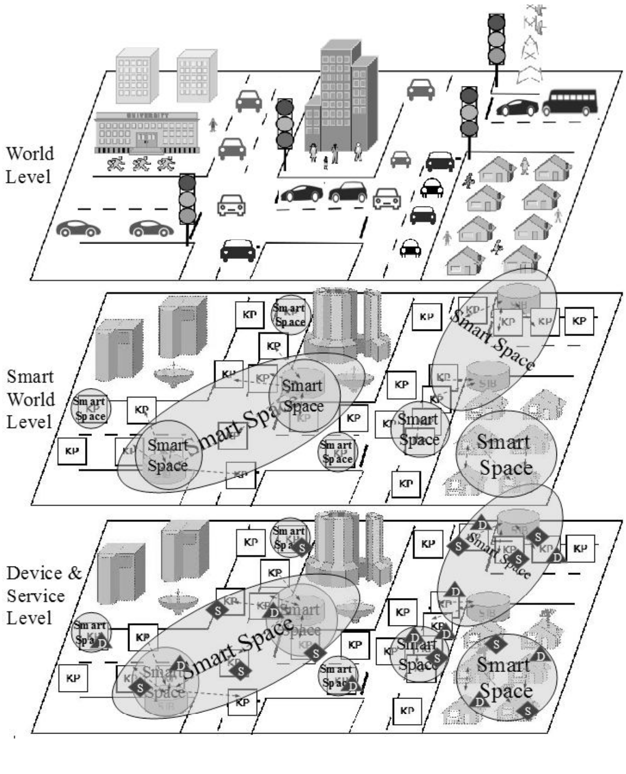

Fig. 1. Three-level conceptual model of logistics service

The conceptual model shows an exemplary scenario of real-world objects logistics service with the mapping of all agents in the Smart Space on the indicated map with traffic and their interactions, as well as smart space brokers (databases). Each space agent generates its own information about a road traffic state (RT) via special sensors or services. Then all the information is collected and analyzed by road space brokers.

RT agents shows as KP squares (cars, buses, pedestrians), Smart Road Information Brokers – cylinders SIB (traffic lights), Smart Space roads presents as oval zones called Smart Space, sensors/devices – a triangle with letter “D”, services – a rhombus with letter “S”, the relationships between agents and spaces are marked with red arrows.

Smart Road can to collect, analyze and process following RT information (the list may be expanded): location, speed (sensor), time at a certain stretch of smart road and other Smart Space agents characteristics.

C. Logistic service scenario description

As a co-option use case is a car fleet management scenario. In this use case systems agents are car fleet owners, drivers and clients. Agents provide information about all car fleet entities (cars, clients orders) and transmit these data for processing in a special logistics service information stores. Each agent may use this information for its own purposes, for example, to assess the car fleet work, its drivers, clients reviews and yet. According to the obtained characteristics is possible not only analysis car fleet data, but also the state of traffic congestion (traffic jams, accidents), the car fleet state, the ability to report about accidents or other traffic accident.

A special knowledge processor (GCSS) is developed for the smart road agents. This KP works within the Smart-M3 platform and it is responsible for receiving and modeling logistics service entities information in Smart Space. All GCSS-related information is placed in a semantic information broker (SIB) which performs analysis, reasoning of new contextual knowledge and push the processed data to a traffic participants. All communication between agents and brokers is performed by using publish-subscribe model [35].

Geo2Tag platform plays a data provider role for service objects (the car fleet owner, drivers, clients) and stores related data in the service database of Geo2Tag platform. Smart-M3 platform is necessary for intelligent services data processing (reasoning of new knowledge), where the GCSS agent is used for projection Geo2Tag service data into the Smart Space in the form of logistics service ontology.

Geo2Tag platform is a primary source of logistics service data, where service users (owner, drivers, clients) fill the service with new data about trips, drivers with available cars in the car park, trips statistics (reviews, events, rating) via the clients mobile applications. Smart-M3 platform uses logistics service data for modeling car fleet information in the Smart Space, its processing and knowledge reasoning.

As a space extension with new data is possible to develop special agents for logistics service objects, for example, a car agent collect relevant information of its characteristics, car fleet statistics agent collect and analysis a general car fleet statistics and other services.

IV. A NEW IMPLEMENTATION OF LBS AND SMART SPACE PLATFORMS INTEGRATION

A. New GCSS programming model

Fig. 2 introduces a new programming model of a Geo2Tag and Smart-M3 platforms integration agent – GCSS.

The logistics service programming model consists of a Geo2Tag and Smart-M3 platforms, as well as the integration agent GCSS with the agents/clients/applications for logistics service (Drivers, Managers, Clients, Cars KP). Additional cloud services can be also connected to platform for storage and further processing of the service data purposes.

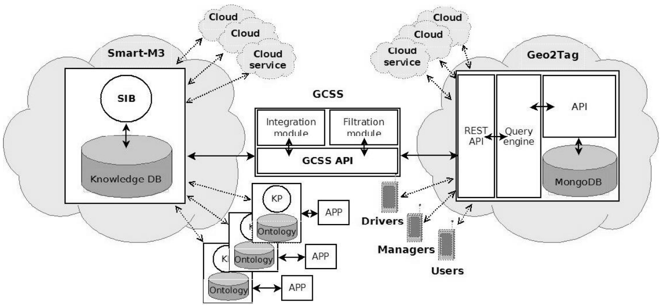

Fig. 2. New software model of logistics service

New GCSS agent will use new version of the Geo2Tag platform [36] with Python API to the Smart-M3 platform. GCSS agent is used for the Geo2Tag platform data projection to the Smart-M3 platform according to the ontological model of the logistic service. As a result, a service space will be filled with all logistics service data through its agents/clients. The ontological model of logistic service is given in chapter 4 C.

B. Logistics service component diagram

Fig. 3 introduces a component model of a logistics service (Geo2Tag, Smart-M3 platforms and GCSS agent). Component diagram consists of three systems – Smart-M3 and Geo2Tag platforms with an integration agent GCSS.

Smart-M3 platform consists of the following main modules (components), such as:

- *transport module* provides access to the platform by various transport protocols (TCP, NoTA, etc.);
- *request handling module* handles the messages of a transport module and translate received data into the platform data exchange protocol – SSAP/XML;
- *graph operations module* is responsible for the representation of received XML messages in the form of a RDF-graph or triples for further processing;
- *triples operations module* provides operations for working with RDF store (insert, query, delete);

- *persistent store module* provides access to the platform knowledge database.

Geo2Tag platform consists of the following main modules (components):

- *platform API module* consists of the set of API for working with platform plugins, services, access control mechanisms and database API;
- *query engine module* handles the received requests and provides access to the platform database (MongoDB [37]) by DB access API;
- *REST API module* is responsible for the providing REST interfaces for a spatial filters, OAuth2 authentication, common access control mechanisms and open data import plugins;
- *MongoDB module* – access to the main Geo2Tag platform data storage for all its services.

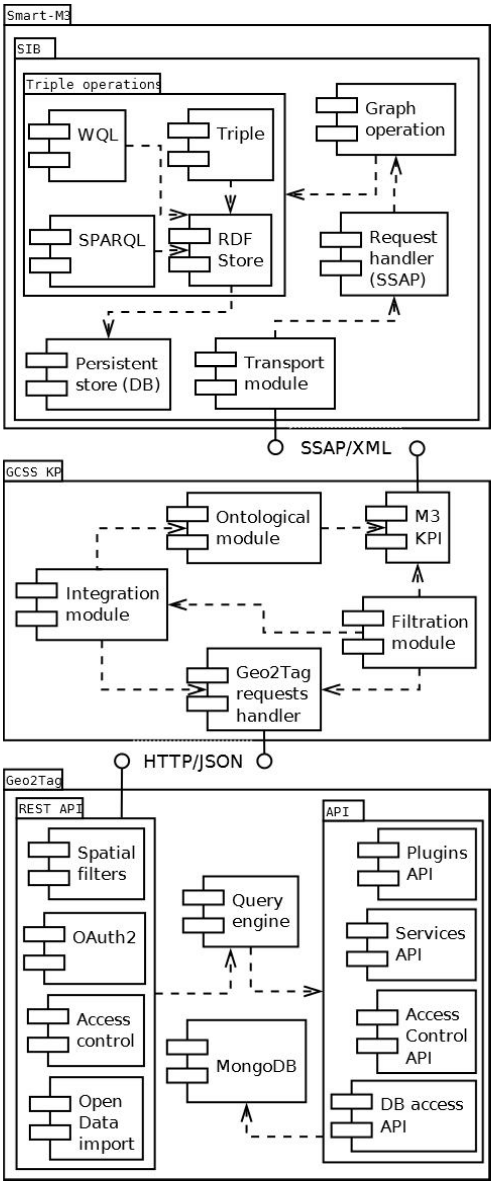

Fig. 3. Component diagram of a logistics service

GCSS platforms integration agent consists of the following main modules (components):

- *Geo2Tag requests hanlder* provides HTTP requests to the main Geo2Tag platform operations (filtration, channels, users);
- *filtration module* is a two-sided data filtration mechanism from the Smart-M3 and Geo2Tag platforms;
- *integration module* converts received geodata into the triplets and vice versa;
- *ontological module* represent converted geospatial data triplets in the form of ontological model [4];
- *M3 KPI* provides API for working with a Smart-M3 platform (connection, triples processing operations, query languages, etc.).

Component diagram shows main modules of the logistics service backend and relations between all system modules.

C. Logistics service ontological model

The ontological model of logistics service is shown in Fig. 4. It consists of the following classes:

- *Owner* – information about car fleet owner (name, phone, email);
- *CarFleet* – car fleet (cars, car brands, the number of free cars, cars in service, statistics);
- *CarFleet statistics* – statistical information about the car fleet (trips number, drivers rating, customer reviews, costs);
- *Driver* – information about car owner, car fleet driver;
- *Driver statistics* – information about driver statistical data (trips number, driver rating);
- *Client* – information about car fleet client (name, phone, address);
- *Client statistics* – statistical information about the client (trip rating, review);
- *Trip* – information about a trip/order (client data, address, cost);
- *Trip statistics* – statistical information about the trip (trip rating).

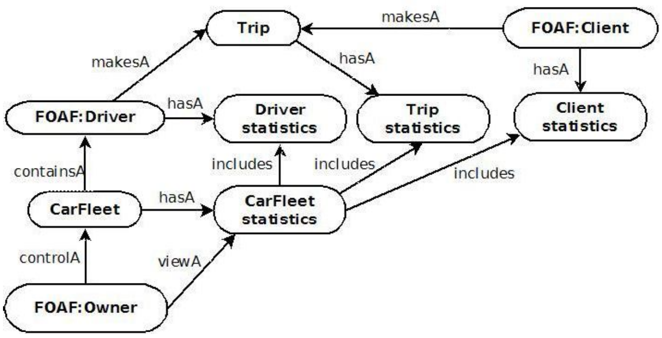

Fig. 4. The ontological model of the logistics service

Main classes of the logistics service ontology are a car fleet owner (FOAF:Owner), a car fleet driver (FOAF:Driver) and a client (FOAF:Client). They contain a constant or rarely changing information and may be described in accordance with a FOAF specification [38]. The car fleet owner may contain one or more fleets, related by properties *controlA*. As an owner (FOAF: Owner), the car fleet class (CarFleet) may also contain one or more drivers (FOAF: Driver), associated with drivers by properties *containsA*. Car fleet contains statistics data (class CarFleet statistics), its owner can browse it data through the property *viewsA*.

Class FOAF:Driver has its own statistical data (class Driver statistics), related by properties *hasA*. The driver makes one or more trips (class Trip) and related by properties *makesA*. The trip has also its own statistics (class Trip statistic). Class FOAF:Client also makes one or more trips (Trip), who have their own statistics (class Client statistic).

In the logistics service ontology can also be added the following classes that will provide to describe the work of the car fleet and its things in more detailed view:

- *car* – information about a driver car and its characteristics;
- *order* – information about a client order (trips);

Logistics service ontology may include any other object/class, which may be associated with logistics service objects. Ontology is used to represent logistics service data as a hierarchy of classes and relations between them. Logistics service ontology is stored in the Smart-M3 platform as a set of RDF triples [39].

D. Logistics service agents

Individual agents implement main car fleet service functionality and perform their tasks to accomplish a common result. Service agents are divided according to the platform and functional.

Geo2Tag platform agents are:

1) a service of the car fleet owner; A mobile agent for creating and managing logistics service car fleet;
2) a service of the car fleet driver; A mobile agent for a car fleet driver and his trips;
3) a client application for a users of the service;
4) a service web page – an actual information about the work of the service.

Logistics service Geo2Tag agents perform all operations for configuring and managing car fleet. These agents are used for collecting and filling services with data about a specific car fleet, its drivers, their travel and clients. After these steps data is represented in the Smart Space. Geo2Tag agents perform following operations:

1) logistics service presetting;
2) call and order fulfillment;
3) browsing statistical data about car fleet orders.

The logistics service preset is a process of creating a new service car fleet and a new drivers invitation with their own and car information. It is follows:

1) a car fleet owner creates a new service car fleet;
2) a driver installs a service driver application on his smart phone;
3) a driver enter his personal data and data characteristics about his car;
4) a car fleet owner invites drivers to the service car fleet, as a car fleet member;
5) steps 2-3 are repeated for all car fleet drivers.

A call and orders fulfillment is a creation of a new trip, selecting a driver or client order and reviews trips and orders. For example, a driver order fulfillment consists of the following steps:

1) a driver launches logistics service application and selects an orders identifiers;
2) a driver presses a button “Start order”;
3) a driver makes an order;
4) in the case of unexpected circumstances that prevent to perform an order, a driver presses to the "Report a problem" button:
    1) displays a pop-up window with a list of possible problems, accidents, technical fault, traffic jams and other problems;
    2) a driver presses to "Send" button.
5) at the end of an order, a driver presses to the "Finish order" button.

A browsing statistical data about orders is carried using a mobile application/agent or a car fleet statistics web page. Statistics include all information about orders, drivers, problems and clients feedback. A description Scenario view statistics consists of the following steps:

1) a car fleet owner opens in a mobile application (on a web page) “Car fleet statistics” screen;
2) the screen displays a table with all drivers and their information: his name, car brand, performed trips today/total, the passed way today/the whole, the need in car service, accidents (by category) today/total and the "Current Location" button to search a driver;
3) a car fleet owner presses drivers "Current Location" button;
4) the screen will appear with “Drivers locations”:
    1) a map with marks of drivers position;
    2) a driver, whose button was pressed, highlighted and map is centered on his location.

All car fleet data and its statistics are stored in the service database of the Geo2Tag platform, this data may be received and processed properly via HTTP/REST [40] queries. The process of modeling Geo2Tag platform information to the Smart Space consists in following steps:

1) reception service data using a Geo2Tag REST API;
2) modeling data to the Smart-M3 platform format (RDF-triples);

3) pushing modeled data into Smart Space according to a service ontology.

The main agent from the Smart-M3 platform side is a platform integration agent GCSS. He used to work with Geo2Tag platform data and modeling it to a Smart Space service ontological model. The additional agent can be a car on-board computer agent (car specifications), a drivers/owner/clients statistics agent for the collection and processing their data in the Smart Space.

For knowledge representation and reasoning of logistics service data responds a Smart-M3 platform, which has a built-in mechanisms to reasoning new knowledge. Its main task is a modeling and reasoning a new knowledge based on the obtained logistic service data.

For example, the logistics service problem is monitoring the state of the driver's cars (average gasoline consumption, car service rate), a general analysis of the collected statistics of a day/week/month and other tasks. A knowledge reasoning mechanisms will be used for the prediction and notification car fleet drivers about the state of his car, possible future trips and so on. Car feet manager will be able to monitor the car fleet autonomous operation, where for orders distribution, current car fleet statistics output and other useful features will engage his personal assistant in the form of an intelligent agent of a logistics service.

V. Conclusion

Sharing the Geo2Tag and Smart-M3 platforms allows you to create a better and more functional solutions for the current problems. Smart-M3 platform with GCSS agent complements the Geo2Tag platform services and allows you to reasoning Geo2Tag services data in a Smart Space.

At present time is carried GCSS agent rewriting using Smart-M3 Python API for working with a new version of the Geo2Tag platform. A further tasks will be to test the basic methods of a GCSS integration agent and the creation of logistics service inference rules for new knowledge based on the Smart-M3 SparQL API [41] or otherwise (SWRL rules or some reasoning frameworks such as FaCT++ reasoner or another) [42, 43].

On the Geo2Tag side are working on the design of geodata visualization interface that allow to implement a high-performance rendering of large amounts of a geodata. Moreover these interfaces will be able to automatically select map scale and center coordinates for the purpose of automatic following on the coordinates of the moving object. In addition, it plans to create a Smart-M3 data import plugin to Geo2Tag platform services. This plugin will be useful not only for creating the service described in this article, but also become a powerful tool for integration of two platforms.

The main planned logistics service functions are the car fleet creation and management with the ability to transfer its data to the Smart Space, its analysis for a reasoning new knowledge of the car fleet work and possibility to make a decisions to perform a service functions.

As further work is planned to study and develop advanced functionality for the "Logistic Service" scenario via OGC standards (OGC SWE Common Data Model, OGC SWE Service Model, OGC GeoSPARQL [44, 45, 46]) and new data sources for the platforms in the form of import plugins to the Geo2Tag platform and agents (KP) or Geo2Tag services for the Smart Space embedded devices (sensors) and other third-party services.

Acknowledgment

The authors would like to thank Finnish Russian University Cooperation in Telecommunication Program for provided support. Some parts of this paper have been prepared within the scope of project part of the state plan of the Board of Education of Russia (task # 2.136.2014/K).

References

[1] Location-based Services – https://en.wikipedia.org/wiki/Location-based_service

[2] D. J. Cook and S. K. Das, "How smart are our environments?" an updated book look at the state of the art, Pervasive and Mobile Compution 3(2), 2007, pp. 53-73

[3] S. Balandin and H. Waris, "Key properties in the development of smart spaces," in Proc. 5th Int'l Conf. Universal Access in Human-Computer Interaction. Part II: Intelligent and Ubiquitous Interaction Environments, UAHCI'09. Springer-Verlag, 2009, pp. 3–12.

[4] K. Krinkin, K. Yudenok. "Geo-coding in Smart Environments: Integration Principles of Smart-M3 and Geo2Tag". In Proceedings of the 13th International Conference, NEW2AN 2013 and 6th Conference, ruSMART 2013, St. Petersburg, Russia, August 28-30, 2013. Proceedings. Springer 2013 Lecture Notes in Computer Science, p. 107-116. ISBN 978-3-642-40315-6.

[5] Nabian, N., Ratti, C., Biderman, A. and Grise, G.: MIT GEOblog: A platform for digital annotation of space for collective community based digital story telling. 3rd IEEE International Conference on Digital Ecosystems and Technologies, Piscataway, N.J.: IEEE, 2009, pp. 353-358

[6] J. Rishede T. Man L. Yiu.: Effective Caching of Shortest Paths for Location-Based Services, SIGMOD '12, Scottsdale, Arizona, USA, 2012

[7] K. Kolomvatsos, V. Papataxiarhis, V. Tsetsos.: Semantic Location Based Services for Smart Spaces", 2nd International Conference on Metadata and Semantics Research (MTSR), Corfu, Greece (2007)

[8] I. Bezyazychnyy, K. Krinkin, M. Zaslavskiy, S. Balandin, Y. Koucheravyi: Geo2Tag Implementation for MAEMO, 7th Conference of Open Innovations Framework Program FRUCT, Saint-Petersburg, Russia, 2010

[9] J. Honkola, H. Laine, R. Brown, I. Oliver, "Cross-Domain Interoperability: a Case Study", Nokia Research Center, Helsinki, Finland, 2009

[10] J. Honkola, H. Laine, R. Brown, O. Tyrkko, "Smart-M3 Information Sharing Platform", 1st Workshop on Semantic Interoperability in Smart Spaces, 2010

[11] D. Korzun, S. Balandin, V. Luukkala, P. Liuha, A. Gurtov, Overview of Smart-M3 Principles for Application Development, Congress on Information Systems and Technologies (IS\&IT'11), Conf. Artificial Intelligence and Systems (AIS'11). Black Sea Coast, Divnomorskoye, Russia, 2-9 September, 2011, pp. 64-71

[12] Top Fleet Management Software Products – http://www.capterra.com/fleet-management-software/

[13] LoJack® Fleet Management solution powered by TomTom – http://www.lojack.com/Fleet-Management-Solutions

[14] Verizon Networkfleet's GPS Fleet Trackng System – http://www.networkfleet.com/

[15] Uber – https://www.uber.com/

[16] Yandex.Taxi – https://en.wikipedia.org/wiki/Yandex.Taxi

[17] Lyft – https://en.wikipedia.org/wiki/Lyft

[18] Gett – https://en.wikipedia.org/wiki/GetTaxi

[19] Yandex.Transport – https://mobile.yandex.ru/apps/android/transport/
[20] GM OnStart – https://www.onstar.com/us/en/home.html
[21] Ford Sync – https://en.wikipedia.org/wiki/Ford_Sync
[22] Chrysler Uconnect – http://www.driveuconnect.com/
[23] TomTom – https://en.wikipedia.org/wiki/TomTom
[24] Garmin – http://www8.garmin.com/solutions/
[25] Google Maps Transit – https://maps.google.com/landing/transit/
[26] I. Sosunova, A. Zaslavsky, T. Anagnostopoulos, A. Medvedev, S. Khoruzhnikov, V. Grudinin. Ontology-based Voice Annotation of Data Streams in Vehicles. Proceedings of 15th International Conference, NEW2AN 2015 and 8th Conference ruSMART 2015, St. Petersburg, Russia, August 26-28, 2015, pp. 152-162
[27] The connected car and privacy navigating new data issues, FUTURE of Privacy Forum (FPF), 13 November, 2014, pp. 1-12
[28] Era-Glonass – http://glonassunion.ru/web/en/era-glonass
[29] eCall – http://ec.europa.eu/digital-agenda/ecall-time-saved-lives-saved
[30] NG-9-1-1 – https://en.wikipedia.org/wiki/Next_Generation_9-1-1
[31] ETSI – https://en.wikipedia.org/wiki/ETSI
[32] 3GPP – https://en.wikipedia.org/wiki/3GPP
[33] Network-centric warface – https://en.wikipedia.org/wiki/Network-centric_warfare
[34] K. Yudenok. "Geo-Coding and Smart Space Platforms Integration Agent Performance Testing and Analysis". In Proceedings of the 14th International Conference, NEW2AN 2014 and 7th Conference, ruSMART 2014, St. Petersburg, Russia, August 28-30, 2014. Proceedings. Springer 2014 Lecture Notes in Computer Science, p. 62-69.
[35] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," ACM Comput. Surv., vol. 35, pp. 114–131, June 2003.
[36] Geo2Tag GeoMongo – https://bitbucket.org/osll/geomongo, http://test.geo2tag.org/
[37] MongoDB – https://en.wikipedia.org/wiki/MongoDB
[38] FOAF – http://semanticweb.org/wiki/FOAF
[39] RDF – https://www.w3.org/RDF/
[40] REST – https://en.wikipedia.org/wiki/Representational_state_transfer
[41] Smart-M3 SparQL – https://www.w3.org/TR/sparql11-query/
[42] SWRL rules –*http://www.w3.org/Submission/SWRL/*
[43] FaCT++ OWL-DL reasoner – http://owl.man.ac.uk/factplusplus/
[44] OGC standards – http://www.opengeospatial.org/standards
[45] OGC SWE – http://www.opengeospatial.org/standards/swecommon, http://www.opengeospatial.org/standards/swes
[46] OGC GeoSPARQL – http://www.opengeospatial.org/standards/geosparql, https://en.wikipedia.org/wiki/GeoSPARQL