Performance Analysis of a “Smart Space” Integration Agent with a LBS Platform

K. V. Krinkin, K. G. Yudenok
Software Engineering and Computer Applications Department
Saint Petersburg Electrotechnical University "LETI"
Saint Petersburg, Russian Federation
kirill.krinkin@gmail.com, kirill.yudenok@gmail.com

***Abstract*—Spatial context is an essential aspect in the Smart Spaces and Internet of Things concepts, as all physical and virtual objects have their location and spatial relationship with other objects. Geographic context characteristics can be used for effective deployment of smart objects in any described environment. For usage and handling smart space subjects geographical context, a special integration agent have been developed for working Smart Space platform with a LBS system. In this article we will talk about a theoretical performance evaluation of this Smart Space and LBS platforms integration system (GCSS).**

***Keywords*—Smart Spaces, Internet of Things, Context-rich systems, Geo-tagging, Smart-M3, Geo2Tag, LBS (Location-based services), Ubiquitous ID, Context-awareness**

I. Introduction

The main objective of this paper is to model and use geographical context in a Smart Space environments. The basis of the Smart Space concept served a “*Ubiquitous computing*” direction, which was defined by M. Weiser in his work “*The computer of the 21st century*” [1].

It should be noted that this work also relates to the “*Context-Rich Systems*” direction, in other words a further development of context-aware systems, which provide not only relevant information/services based on the users context, but also provide the ability to adjust the methods for obtaining context information, respond on emerging events and provide information or recommendations for further user actions. This work and its technologies stack affects to the "*Internet of Things*" concept as a spatial context is the basis for the design modern services in various physical or virtual environments. According to the research agency Gartner[1], "*Context-Rich Systems*" and "*Internet of Things*" directions are a key strategic technology trends in 2015 year.

Smart Space should provide a continuous distribution of semantic data between all participants of the space and provide a communication field for the services, that can be run on various types of computing environment devices (personal devices, stand-alone computers or robots). Smart Space participants are autonomous agents that designed to monitor the state of the computing environment and make their own changes on the basis of the knowledge gained in the process of monitoring.

The practical implementation of the Smart Space concept requires a variety of models, algorithms, technologies, hardware and software. *V. I. Gorodetsky* identified the following problems in the smart space environments [2]:

1. the problems of integration heterogeneous models, hardware, communication protocols, user interfaces and software into a single system;
2. the need for co-processing of heterogeneous information from distributed sources (sensors, databases, email and web etc.) for the modeling current smart space context, online and offline training, web-services searching, context forecast, solving problems of decision-making and others.

The Smart Space concept is based on the Web 2.0 technology stack (Semantic Web), client-server and distributed service-oriented architecture, context-aware computing, human-computer interaction and others. For modeling and processing smart space data, semantic knowledge representation models are used (RDF, OWL).

The semantic information representation models have their own advantages and disadvantages. Advantages include – integration of heterogeneous data, flexible data schema changes, the possibility of semantic search and joint learning of new knowledge, as well as disadvantages are – systems scalability and performance problems [3, 4, 5]. The scalability problem is solved by the Smart Space platform – Smart-M3 [6, 7, 8].

The performance problems arise when it is necessary to handle large volumes of data, that modeled with a semantic knowledge representation models. To resolve this problem, it is invited to move geographical smart space information processing into a separate platform, LBS-system. The integration of Smart Space platform with a LBS system will also use a variety techniques (search, filtering) for processing geographical context in a Smart Space environments.

In this article we will talk about a performance analysis of a Smart Space Smart-M3 platform integration agent with a LBS system. The article is organized as follows: In the Introduction

1 *Gartner's Top 10 Technology Trends for 2015* – http://www.gartner.com/newsroom/id/2867917

describes main project task, basic problems in a Smart Space concept and offers a solution for the system performance. Chapter 2 briefly describes a proposed solution for the Smart Space Smart-M3 platform performance improvement during processing a geographical information. Chapter 3 provides a response time function analysis of the platforms integration agent and separately processed data volumes of each platform. Chapter 4 describes the performance winnings function when moving geographic information processing to the separate LBS system.

II. Platforms Integration Task

For the smart space platform performance improvement while processing a smart space subjects geographical context, a special platforms integration agent (GCSS) was developed, which is responsible for the Smart-M3 and LBS platforms interaction and provide mechanisms for the transparent modeling and processing data of integrable platforms. Implementation details and testing analysis of the GCSS platforms integration agent are given in [9, 10].

Platforms integration implemented using a «Ubiquitous computing» direction standard – *Ubiquitous ID* [11]. The main element of this technology is a unique 128-bit identifier - *ucode*, which is used for the Smart-M3 and LBS platforms integration.

Assume that each smart space knowledge processor (agent) has it's own unique *ucode* identifier and own LBS system user. Every LBS user can has an unlimited number of data channels to which he can subscribe and receive data. While integration platforms, each smart space agent creates data channels and links them with own *ucode* identifier. If agent know *ucode* identifier, he can access to the smart space subjects geographic context data from the LBS system data channels.

As an example serves the work [12], which deals with the implementation of virtual computing platform for solving the interaction problem of non-uniform devices in a «Ubiquitous computing» and "Internet of Things" directions, that built on a Smart-M3 platform and ideas of *Ubiquitous ID* standard.

From a practical point of view, we use Smart Space Smart-M3 platform and Geo2Tag LBS-system for modeling and processing smart space subjects geographical context.

Smart-M3 platform is an open software implementation of a Smart Space concept via M3 architecture [6]. Abbreviation M3 focuses on properties – multi-device (multiple devices), multi-vendor (many equipment manufacturers) and multi-domain (various set of domains) characteristic for the computing environments in the "Internet of Things" direction. The most essential features of upcoming software are pro-activeness and context-awareness. Context-aware systems are working within a context life cycles. Life cycle consists of specific steps, which are responsible for the sensing, modeling, processing, learning and dissemination of contextual information in any computing environment [13].

As a LBS system stands an open source software platform Geo2Tag [14], the main features of which are responsible for geographic tags, tags channels, platform users management and also provide extensive functionality for two-dimensional and three-dimensional filtering.

The Smart-M3 platform performance is poor, recommendations for its optimization are given in [10]. Geo2Tag LBS system specifically optimized for storing and processing large volumes of geographical information, it's performance is discussed in [15].

Next comes a discussion about a performance analysis of a GCSS platforms integration agent.

III. GCSS Performance Analysis

System performance is an inverse value of the system response time. GCSS system response time calculated by the formula (1), which depends on used platforms and network bandwidth, where each platform consists of its own components.

$$F(t)=\frac{(T_{LTD}+T_{PG}+T_{SL}+BW_{SSAP})}{T_{OP}} \qquad (1)$$

T_{LTD} – Lighttpd web-server responce time, $T_{PG}(\rho, R)$ – PostgreSQL DB response time, $T_{SL}(\rho, R)$ – SQLite DB response time, BW_{SSAP} – Smart-M3 SSAP/XML protocol bandwidth,

$BW_{NET}=\frac{1}{T_{OP}}$ – network bandwidth, where T_{OP} – system operation execution time.

System performance is also dependent on the processed amount of data. A lot of interest in solving the system performance problem is a Smart-M3 and Geo2Tag platforms databases response time, that used by GCSS platforms integration agent.

Let Q – GCSS system query model, which depends on:

$$Q=F(\rho_{xy}, R, N_s) \qquad (2)$$

, where ρ_{xy} – density points per area unit, R – filter radius (LoadTags), N_s – amount of space data.

The amount of Geo2Tag platform data is determined by the filter parameters that used for the extraction of geographic data from the platform database (PostgreSQL). For the two-dimensional *LoadTags* filter by radius – the amount of data N depends on the circle area and the density points per circle area unit – ρ_{xy} and R parameters.

$$P_{PG}^{-1}=F(\rho_{xy}, R)=f(N) \qquad (3)$$

, where $N=\rho_{xy}\pi R^2$;

For the Smart-M3 platform, the amount of space data has a linear dependence, as it depends on the number of triples in the smart space database (SQLite, Virtuoso, Redland). For the smart space that containing subjects geographical data, data volume is calculated by the formula:

$$P_{SL}^{-1}=F(F(\rho_{xy},R)+N_s)=F(f_{G2T}(N)+N_s) \qquad (4)$$

, where $f_{G2T}(N)$ – volume of smart space data, N_s – volume of other smart space data.

By varying ρ_{xy} and R parameters, we can exactly say how much data will be processed by the system and how much time it will take. The processing time surface of the total data amount for the Smart-M3 and Geo2Tag platforms presented at the Fig. 1.

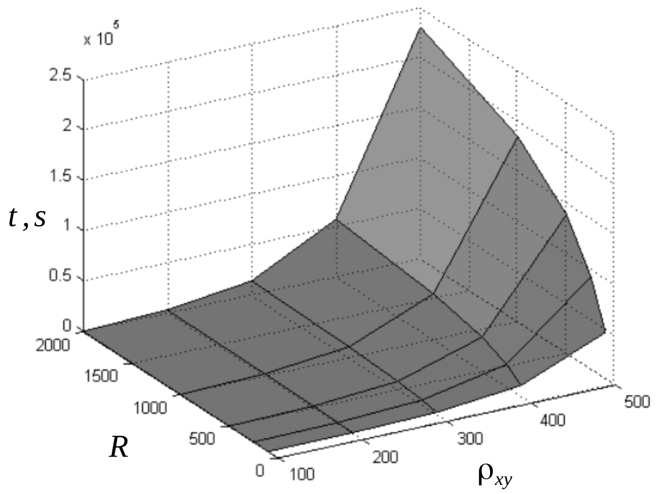

Fig. 1. Processing time surface of *N* data volume by density points and figure radius

IV. GCSS Performance Winnings

Consider the GCSS system performance, that depending on the processed data volume by the Smart-M3 and Geo2Tag platforms.

Let's *SQLite(SM3)* is a number of the smart space triplets and *PostgreSQL(G2T)* is a number of geographic tags (geotags) presented in the *{time, latitude, longitude, altitude, data}* format. It turns out that each geotag described in a smart space by four triplets for the geotag fields – *time*, *latitude*, *longitude*, *altitude* (without field *data)*.

Assume that Smart-M3 platform data processing time depends on:

1. the number of the smart space triplets;
2. a smart space platform processing delay time.

$$T_0=SQLite(N_{triples})+T_{SM3}(Triples) \qquad (5)$$

where T_{SM3} – Smart-M3 platform delay time.

A total volume of smart space data is a sum of all smart space triplets:

$$N_{AllTriples}=N_{GeoTriples}+N_{OtherTriples}; \qquad (6)$$

Operating (response) time of a Smart-M3 platform increases with a number of triplets, because the amount of space data has a linear dependence.

The general time formula for the processing a certain amount of a smart space data is a sum of the times spent on processing:

1. not geographic space triplets – $\neg Gtr$;
2. the delay for processing not geographic triplets;
3. geographic space triplets – N_{Gtr} .

$$T_1=T_{SL}(\neg Gtr)+T_{SM3}(\neg Gtr)+T_{G2T}(N_{Gtr}) \qquad (7)$$

, where $N_{Gtr}=F(\rho_{xy},R)$

Lets calculate a total data processing time by the Smart-M3 platform and a total amount of smart space data with a subjects geographic information.

Define N_{SQSM3} – number of all space triples.

$$N_{SQSM3}=N_{SQSM3Triples}+N_{SQSM3GeoTriples} \qquad (8)$$

The total smart space data processing time consists of the Smart-M3 platform database response time and platform processing delay time:

$$T_{SM3}=T_{SQ}(N)+T_{SM}(N) \qquad (9)$$

The total geographic data processing time of Geo2Tag LBS system consist of a Geo2Tag PostgreSQL database response time and platform processing delay time:

$$T_{G2T}=T_{G2T}(N)+T_{PG}(N) \qquad (10)$$

Therefore, the total smart space data processing time without a geographic context is $T_O^1=T_{SM3}$ and a total smart space data processing time with a geographic context is:

$$T_O^2=T_{SQ}(N_{SQM3Tr})+T_{SM}(N_{SQM3Tr}) + T_{G2T}\left(\frac{N_{SQM3Gtr}}{4}\right)+T_{PG}\left(\frac{N_{SQM3Gtr}}{4}\right); \qquad (11)$$

A system performance winnings by transferring geographic context processing to a separate platform (LBS-system) is:

$$P_{Win}=T_O^2-T_O^1 \qquad (12)$$

Fig. 2 shows the GCSS system performance winnings when moving smart space geographic information processing to the separate LBS platform (Geo2Tag). The upper surface is equal to the surface of the total platforms data volume (Fig. 1), middle surface – performance winnings surface when moving geographic data processing into the LBS system, bottom surface – the number of smart space data without subjects geographic information.

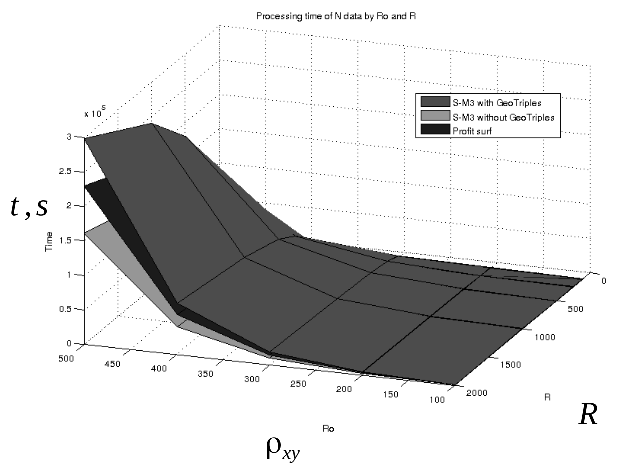

Fig. 2. System performance winnings surface (black color)

CONCLUSIONS

The total amount of smart space data formed from the set of all space triplets. For the representation of smart space subjects geographical information in a geotag *{time, latitude, longitude, altitude, data}* format, you need to create five additional triplets for each smart space subject. And since the smart space data amount has a linear dependence, it is bad for the overall system performance, and it turns out that the smaller data we have in the space, it is better for the smart space platform performance (Smart-M3). Therefore, the geographic information transferring to the separate platform (LBS system) will increase the smart space system performance proportional to the geographical data volume, that have been moved from the space where the overall GCSS system performance is bounded above the Smart-M3 platform performance.

REFERENCES

[1] M. Weiser. "The Computer for the 21st Century", Scientific American Special Issue on Communications, Computers, and Networks, September, 1991.

[2] Gorodetskiy V.I. [Agents and knowledge extracting from data in the smart space]. Sbornik trudov pervoy mezhdunarodnoy konferentsii «Avtomatizatsiya upravleniya i intellektual'nye sistemy i sredy» [Proceedings of the First International Conference «Control Automatization and Smart Systems and Environments»]. Terskol. 2010. pp. 24–36. (In Russian).

[3] Richard Benjamins, Jesus Contreras, Oscar Corcho, Asuncion Gomez-Perez. "Six Challenges for the Semantic Web". In Proceedings of KR2002 Semantic Web Workshop, 2002.

[4] Bhavani Thuraisingham. "Security standards for the semantic web". Journal of Computer Standarts and Interfaces. Volume 27 Issue 3, March 2005, pp. 257-268.

[5] Gopal Pandey. "The Semantic Web: An Introduction and Issues". International Journal of Engineering Research and Applications (IJERA), ISSN: 2248-9622, Vol. 2, Issue 1, Jan-Feb 2012, pp. 780-786

[6] J. Honkola , H. Laine, R. Brown, O. Tyrkkö, "Smart-M3 Information Sharing Platform", Proc. IEEE Symp. Computers and Communications (ISCC'10), IEEE Computer Society (June 2010) 1041–1046.

[7] 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.

[8] D. Korzun, S. Balandin, V. Luukkala, P. Liuha, A. Gurtov, "Overview of Smart-M3 principles for application development", in Proc. Congress on Information Systems and Technologies (IS&IT'11), Conf. Artificial Intelligence and Systems (AIS'11), 2011/9, pp. 64-71.

[9] K. Krinkin, K. Yudenok. "Geo-coding in smart environment: Integration principles of Smart-M3 and Geo2Tag" in Proc. 13th Int'l Conf. Next Generation Wired/Wireless Networking and 6th Conf. on Internet of Things and Smart Spaces (NEW2AN/ruSMART 2013), ser. LNCS 8121, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds. Springer-Verlag, Aug. 2013, pp. 107–116.

[10] K. Yudenok. "Geo-coding and Smart Space Platforms Integration Agent Performance Testing and Analysis" in Proc. 14th Int'l Conf. Next Generation Wired/Wireless Networking and 7th Conf. on Internet of Things and Smart Spaces (NEW2AN/ruSMART 2014), ser. LNCS 8638, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds. Springer-Verlag, Aug. 2014, pp. 62–69.

[11] N. Koshizuka, K. Sakamura. "Ubiquitous ID: Standarts for Ubiquitous Computing and IoT". Pervasive Computing IEEE CS, 2010, pp. 98-101

[12] J. Kiljander, A. D'elia, F. Morandi, P. Hyttinen, J. Takalo-Mattila, A. Ylisaukko-Oja, J.-P. Soininen, T. S. Cinotti, "Semantic Interoperability Architecture to Pervasive Computing and Internet of Things". IEEE Access Volume 2, 27 August 2014, pp. 856-874.

[13] C. Perera, A. Zaslavsky, P. Christen, D. Georgakopoulos, Context Aware Computing for The Internet of Things: A Survey, IEEE Communications Surveys and Tutorials, Volume PP, Issue 99, pp. 1–44, 2013.

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

[15] M. Zaslavsky, K.Krinkin. "Geo2tag Performance Evaluation", Proceedings of the 12th Conference of Open Innovations Association FRUCT and Seminar on e-Travel, Oulu, Finland, 2012.