An Architecture Proposal for the Creation of a Database to Open Data related to ITS in Smart Cities

Sérgio A. A. Barbosa
Faculdade de Administração de Negócios de Sergipe (FANESE)
Sergipe, Brazil
saabarbosa@gmail.com

George Leite Júnior
Instituto Federal de Sergipe (IFS)
Sergipe, Brazil
george.junior@ifs.edu.br

André S. Oliveira
Instituto Federal de Sergipe (IFS)
Sergipe, Brazil
andre.stos.oliveira@gmail.com

Telmo O. de Jesus
Instituto Federal de Alagoas (IFAL)
Alagoas, Brazil
jtelmooliveira@yahoo.com.br

Dr. Douglas D. J. de Macedo
Universidade Federal de Santa Catarina (UFSC)
Santa Catarina, Brazil
douglas.macedo@gmail.com

Dr. Rogério P. C. do Nascimento
Universidade Federal de Sergipe (UFS)
Sergipe, Brazil
rogerio@ufs.br

ABSTRACT

As a result of population growth in large cities face everyday problems related to urban mobility such as congestion, quality of urban roads and inefficiency of public transport. Intelligent transport systems initiatives act as an efficient solution to improve the functioning and performance of traffic systems, reducing congestion and increasing safety for citizens. However, due to the inclusion of different and distributed information sources on urban mobility, interoperability of the various technologies involved and the retention of these data are challenges that involve complex and costly efforts to governments and businesses. Thus, this article presents a proposal for georeferenced data retention architecture of Intelligent Transportation System in order to store this information georeferenced urban mobility in order to allow perform these activities more easily, and to promote interoperability between various applications. Therefore, the proposal was i9ITS architecture based on Service Oriented Architecture. It conducted a case study related to building an application that uses the i9ITS architecture for a taxi service company with real data. The use of this architecture proved to be effective and efficient to meet the proposed problem, as well as other possibilities to meet the demands and challenges related to Intelligent Transportation System.

CCS Concepts

•**Information systems** → **Data management systems;** Middleware for databases; **Service buses;**

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

Keywords

smart cities, intelligent transport systems, service-oriented architecture, open data, enterprise service bus

1. INTRODUCTION

With the evolution of societies and the necessity of agility and efficiency when it comes to the access to services and information, the connectivity takes an essential role on this global and digital interaction. In this context, the smart cities gain space and present themselves as a viable path, as it was observed in studies which point out to the utilization of technologies in an urban context [10].

The connectivity inherent to the smart cities opens a very promising border with regard to the control of access to information [11] [18] and architectures distributed to intelligent transportation systems [13].

According to the IPEA (Instituto de Pesquisa EconÃt'mica Aplicada), one of the problems of the urban centers, nowadays, is urban mobility. The reflections on urban transportation are evident, characterized specially by an increase of traffic on the city roads and consequently an increase of traffic jam situations [1].

An ITS (Intelligent Transportation System) represents [13] "the application of advanced sensors, computers, electronic devices and communication and integrated strategic management technologies aiming at the improvement of security and the efficiency of the traffic management system".

The sources of TIC (Technology Information e Communication) generated by ITS form complex structures and produce a great volume of data, which present great challenges and opportunities, hindering the possibility of providing software which integrate information from sensors and which collect the physical space in real time. A possible approach to assist in the integration problem is the use of open data.

Open data can be understood as data which can be used freely by anyone. It is worth to highlight that open data initiatives are being realized by governments in all the levels, with the objective of increasing the transparency, capacitating properly the citizens, fomenting innovations and repairing public services [11]. These initiatives converge with smart cities and diverse solutions are already a reality in cities like New York, Amsterdam, Helsinki, Chicago, Barcelona, Quebec City, Rio, Dublin, Nairobi and Manchester [11].

It is important to consider the importance of approaches in architectures of software systems to receive data related to information of urban mobility georeferencing, online form or batch, as well as an alternative to solve the lack of standardization of the formats in the receiving of such sources of data.

Inside this context, this article presents a proposal which use the paradigm SOA, which plans to receive information from urban mobility georeferencing available by intelligent transportation systems, through Webservices, to form an open urban mobility database. To verify the efficacy and efficiency of this proposal, it was performed a case study with data from a taxi service company in the city of Aracaju (SE) âĂŞ Brazil.

This paper is organized as follows: in section 2 are described the works related to the themes of this study, in section 3 is mentioned the theoretical framework where relevant concepts and definitions are described, in section 4, information about the i9ITS architecture, in section 5, the case study that verifies the effectiveness and efficiency of the architecture proposed in a real application, in section 6 is made an analysis of the data and discussions obtained in the case study and finally, in section 7, the conclusion of this study, as well as the possible future work that could be realized.

2. RELATED WORK

Diniz, Silva and Gama [7] propose reference architecture, using shelf components (off-the-shelves), which implements a middleware platform for Crowdsensing solutions in smart cities. The proposed solution has as characteristics: being configurable for multiple domains, using complex event processing for data analysis in real time, integrating sensors through services (implementing SOA through RESTful Webservices using Javascript Object Notation (JSON)) and asynchronous messages (using event-driven architecture - EDA), as well as the use of human sensors through the use of Crowdsensing. They built an Android application for bike traffic accidents record for performing experiments in order to check the resistance and stability of the proposed approach and after the performed tests, they concluded that the results were satisfactory. However, this work depends entirely on data sharing through a participatory application, which involves the population in the collection of information.

Quintero et al. (2012) , use the WSN (wireless sensor networks) technologies and SOA for creating an application of ITS. The application consists of the location of unoccupied parking spaces in the University of Alicante (Spain). The authors mention that SOA has been used for integrating information in traffic centers, as well as in the integration of services for public transport and point out that both technologies were presented as suitable solutions for ITS. The combination of these technologies obtained better results with regard to safety, making possible the use in other ITS applications such as traffic management, public transport, fleet management, among others.

Li (2010) designs a system of TIS (Travel Information Service) distributed, based on SOA that enables service providers (bus companies, subway, train, gas stations) to share and retain their own resources in the platform. This system consists of four parts: traveler, integrated platform, TIS record and the service provider. The traveler shall no longer request information to the service providers directly, but to the integrated platform. The platform is responsible for identifying the necessity of the user to transform it into a certain type of service to meet the needs of the traveler. Then, the platform will call the appropriate services and synthesize the response results for the traveler. A simulation requesting a service in the city of Beijing was held. This approach has focused only on public transport services.

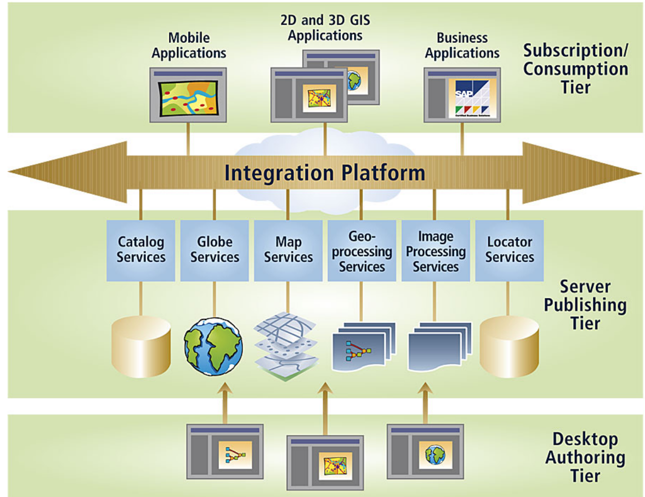

Figure 1: Example arthitecture SOA [4]

3. THEORETICAL BEANCHMARK

3.1 Service Oriented Architecture

In the literature, there are several definitions of SOA. In the context of this paper, SOA is defined as a paradigm of software architecture as well as IT infrastructure that supports and facilitates interoperability of data and distributed applications in diverse technologies [6]. Services are activities to perform business processes, defined as reusable and independent that encapsulate features which add value to those who use them [6]. Services are available on the Enterprise Service Bus (ESB) component, which, through it, it is possible to connect services and applications that will consume them.

There are several benefits of using an ESB [6], such as low coupling, well-defined service contracts, service abstraction, autonomy, reuse, composition and ease to be found, figure 1.

The types of services to be provided will depend on the application requirements and its technologies. The most common are [6]: Webservices and Representational State Transfer (REST).

3.2 Intelligent Transportation System (ITS)

ITS aims to support various everyday situations related to urban mobility, through the use of technologies and interoperability between communication systems, data trans-

Figure 2: Architecture Elements ITS [14].

Figure 3: Open data relationship information [9].

mission and connectivity.

Its efficiency in the monitoring and agility in the distribution of information are essential for the results, with regard to the optimization of transport systems, to be felt by the general population that is part of this scenario which, as a result, allows a better management of the urban transport system [18] [13].

ITS [13] can be subdivided into six advanced management areas: Advanced Traffic Management System, Advanced Information Systems for Travelers, Public Transports Advanced Systems, Commercial Vehicles Operation Systems, Advanced Vehicle Control Systems, Electronic Toll Collection System.

All these subsystems, figure 2, are intended to act, in a targeted and specific form, upon sub-areas of transport management, seeking to ensure the efficiency and quality of urban mobility.

3.3 Open Data

According to the Open Definition [2], open data are data that can be freely used, reused and redistributed by anyone - subject, at most, to the requirement of attribution to the original source and sharing under the same licenses in which the information was presented. In other words, the opening of data is interested in avoiding a mechanism of control and restrictions on the data that are published, allowing both individuals and legal entities to exploit this data freely. From this perspective, the definition of open data presents three fundamental rules [5]: availability and access, reuse and redistribution and universal participation (areas of expertise and people / groups).

From the moment that there is a move to open data, where the three aforementioned core standards are respected, it is possible that different organizations and systems can work collaboratively. This happens due to the ability of these organizations and systems to interoperate data that were opened, expanding the communication and enhancing the efficient development of complex systems [8].

Ojo and Zeleti (2015) relate smart cities with open data. This relationship is performed through specialized data (transportation, education, health, etc.) and dynamic ecosystems, as open data impact on smart cities when there is impact of mastery and open government data as shown in figure 3.

4. PROPOSAL OF I9ITS ARCTHITECTURE

The architecture called i9ITS, use the SOA approach and aims to facilitate interoperability between systems in an automated and manageable way whose purpose is to enable the entry and standardized collection of geo-referenced data in order to build an open database of urban mobility.

4.1 Overview

Therefore, some requirements must be observed in the construction of this architecture proposal: the use of data from covenant companies; consolidation, only of the relevant data to urban mobility; creation of a structure to store georeferencing data and consideration of data privacy.

The first requirement is to formalize agreements between companies wishing to use the proposed architecture. Through this agreement will be set up a restricted access so that these companies can insert data in the database.

The second requirement is aimed at defining the capture mechanism of the relevant data to urban mobility, excluding data that are not related to this context.

The third is to create a database capable of storing data related to georeferencing connected to urban mobility in order to allow the consolidation of this information for future provision of access on an open database.

The last requirement comprises the corresponding activity to the process of the overshadowing of identity of the georeferenced element to be inserted. The goal is to ensure data privacy avoiding the identification of the object corresponding to that coordinate. For this, will be used a conversion algorithm based on Message-Digest algorithm 5 encryption mechanisms (MD5 Hash Generator) to be applied on the identifiers given in the reception of the data. This step becomes necessary because the information is not derived only from government agencies, like most open data applications in the literature, so, for security reasons the monitored object must not be named.

4.2 i9ITS Architecture

The third-party geo- referencing applications (Fig. 1 (1)) collect geographic coordinate information from various sensors that can be grouped and treated to provide important sources of information on urban mobility.

Considering this scenario, the i9ITS architecture comes as a proposal which predicts the reception and retention of such information, by using the SOA approach, using input methods, available by Webservices RESTful, to the services of Back-End module (Fig. 1 (2)).

The Back-End module (Figure 1 (2)) defines the input methods through RESTful Webservices (Figure 1 (2.a)), which is available in the catalog of services through the Enterprise Service Bus (ESB) middleware that coordinates the access to the synchronous services of Request / Response of the ITS application (Figure 1 (2.c)). Each ESB Webservices input method (Figure 1 (2.a)) is concerned with treating the types of processing, online or batch, as well as the definition of inputs of the geo-referenced data via NMEA (National Marine Electronics Association) pattern or not.

The sendLocalizationBatch method aims to deal with the batch type processing, by importing georeferenced data where it receives a file following a pre-defined layout pattern. The information outlined in this layout are: Token (text), Company Identifier (text), Vehicle Identifier (text), Latitude (decimal coordinate), Longitude (decimal coordinate), Altitude (numeric), Speed (numeric), Date / Hour (yyyyMMddHHmmss - year month day hour minute second), Vehicle Type (entire) and Protected (logical). The information of the Company identifier, the Vehicle, Latitude, Longitude and Date / Time are mandatory. For the Type of Vehicle will be allowed the following information: 0 - Private Car, 1 - Taxi 2 - Bus, 3 - Collective Bus, 4 - Train 5 - Metro, 6 - Bicycle, 7 - Truck, among others.

For the online processing will be available the methods: sendLocalizationNMEA and sendLocalizationGeneric. The sendLocalizationNMEA method uses the standard NMEA [12] which defines communication of navigation electronic devices, through the GGA sentence, which contains 3D global positioning information (Latitude, Longitude and Altitude) [16]. The sendLocalizationGeneric method will be used to receive data from other devices that do not adopt the standard NMEA such as smartphones or OnBoard Unit (OBU) in VANETs needing to use the same batch file format standard, however, containing only a string of transmission.

To prevent unauthorized data input at the base, an authentication has been set for access to the ESB methods to be carried out by covenant companies, it was used the cross-domain authentication model [17]. This model defines a trust relationship between the client and the ESB, via a token to be sent to each call of the methods. The ESB is responsible for providing the method authenticationService however, is the responsibility of the Safety Application (Figure 4 (2b)), perform the authentication and token generation that will be passed on to the ESB and forwarded to the client application.

All methods provided by the ESB also have the token as a parameter, which is generated by authenticationService service (Figure 1 (2.a)). AuthenticationService receives the identification data from the convened company and returns the token which must be informed in the call of the sendLocalizationBatch, sendLocalizationGeneric and sendLocalizationNMEA methods.

The ESB component (Figure 4 (2.a)) is responsible for the process of authentication of the supply company and validation of the number of parameters, making the request

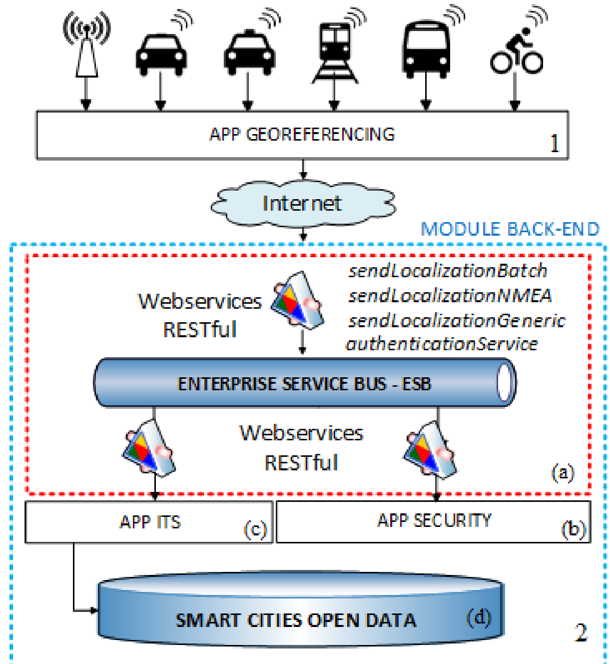

Figure 4: i9ITS Architecture (Source: Authors).

to be passed to the ITS application only if valid. It was also defined the obfuscating of the forwarded information (Company Identifier and Vehicle Identifier), transforming that data into encrypted information, thus ensuring the privacy requirement.

The ITS Application component (Figure 4 (2.c)) aims to address and consider the data received on the field of ITS, and to persist them in smart cities open data base (Figure 4 (2.d)).

5. CASE STUDY

The case study evaluated the effectiveness of the proposed architecture through the use of real geo-referenced data provided by a taxi service company in the city of Aracaju (SE) - Brazil.

Based on i9ITS architecture proposal, the Mule ESB middleware was used because it is an Enterprise Service Bus open source, it is extensible and has free version. For the HTTP data communication between the server and client applications it was used the JSON format implemented on the RESTful style because of its performance and simplicity.

A service was made available through the url [1] in order to receive the request from georeferencing applications suppliers, which will hold the process of dazzling of the referenced object identifier, and finally, the data will be sent to the application of ITS which will make the validation for the insertion into the base.

The process is initiated with the HTTP request (Figure 2 (1)) in which the obtained data are sent by the URL (token, idCompany, idVehicle, latitude, longitude, altitude, speed, dateTime, typeVehicle, protected) to the SetPayload component (Figure 2 (2)), then, these data are submitted to Java class (Figure 2 (3)), which aims to validate the token and

[1] http://www.i9frota.com.br:6061/sendLocationGeneric

overshadow the id of the Vehicle and the company through the MD5 hash if the Protected parameter is true. After obfuscation, it is generated a string in JSON format to be sent to ITS Applications.

The result of the processing of the class is stored in the variable component (Figure 2 (4)). If the result returned by the JSON (Figure 2 (5)) is "error": true (Figure 2 (6)) it indicates that there was error in the processing and the flow is interrupted displaying the Failure message. Otherwise, the data are validated through a Java class (Figure 2 (7)) and the result of processing is stored in the Variable component (Figure 2 (8)). If the result returned by the JSON (Figure 2 (9)) is "error": true (Figure 2 (10)) it indicates that there was error in the processing and the flow is stopped displaying the Failure message. Otherwise, it is performed a request for the APP ITS application (Figure 2 (11)) passing the generated JSON, which will store in the open base returning the Success message.

This case study sought to validate the architecture using real data through a vehicle tracking system from a taxi company, which detects the location of 120 vehicles (each transmitting a coordinated georeferenced at a frequency of 30 seconds) for 15 consecutive days.

This taxi company manages requests through a cab monitoring system (Figure 6). These requests are collected and directed respecting the order of the first taxi driver in line in their respective mapped areas. The screenshot of the system displays the coordinates of the plotted vehicles in a specific area of the city of Aracaju (SE) - Brazil.

The objective was to verify the complexity in the change in the solution of the vendor to perform the integration and also to analyze the possible loss of performance. The change was made in the application of the supplier with the calling of the sendLocationGeneric method from the API client of the i9ITS architecture made in Java, in order to send the data online so that the open base represents the real-time information. The test environment was divided into three different servers in the cloud (hosting Server4you cloud), being two physical and one virtual.

For the assembly of the environment, Mule ESB Standalone 3.7 was used on one physical server whose configuration is described on Table 1.

As a repository of information on smart cities open data (Figure 4 (2c)), was used DBMS PostgreSql 9.1 with PostGIS extension in order to store the use of geo-referenced objects, installed on a virtual server whose configuration is present in Table 1. In this repository were created two tables: OBI_OBJECTID (stores the identifications of vehicles) and MOB_MOVE_OBJECT (stores the movements of vehicles).

For the execution of ITS App was used the physical server described in Table 1.

6. ANALYSIS OF RESULTS AND DISCUSSION

The analysis of results was performed from two perspectives: effectiveness and efficiency. On the effectiveness it was verified if the quantity of performed transmissions were successfully persisted in the smart cities open data database. With regard to efficiency, it dealt with the measuring of the time of each request on the basis of a successful metric as performed by the authors in [3].

Table 1: Configuration Servers Involved in the Case Study

Server	Operational System	RAM	Link
ESB Mule Standalone	Ubuntu 12.04 LTS - LAMP	8 GB	100 Mb/s
SGBD	VM -Ubuntu 10.04 - Plesk 11	2 GB	100 Mb/s
App ITS	Windows Server 2008 R2 Standard SP1	4 Gb	100 Mb/s

To perform the analysis of architecture from the perspective of effectiveness, consultations were held in the smart cities open data database, comparing the quantity of requests to the total of daily persisted records by georeferencing application from the provider. The obtained result indicated that all requests were successfully stored.

From the perspective of efficiency, it was obtained from the library client of the i9ITS API, the time of each request made and calculated the daily average. It was taken as a metric the analysis of the time of request where values less than or equal to 169 milliseconds will be considered "Success". The "failure" was considered taking as a basis the sum of the daily standard deviation and the daily average, in other words, requests times greater than this threshold. Table 2 shows the results obtained through this analysis.

Table 2 shows that during the days of data collection were identified failures on an average of 2,16% in requests made between the application of georeferencing and i9ITS architecture. Figure 7 shows the result obtained on the percentage view of the requests made, stressing that the success percentage was satisfactory in relation to the number found in [12] where it was made a comparison between webservices requests in different programming languages having the time of Java ranging between 50 and 150 ms.

It was also observed that there was no record of increase of processing in the application of the supplier. The data obtained demonstrated that all requests sent by Georeferencing Application were processed correctly at an average time of 123 milliseconds. The failure rate, in average, reached a value of up to 3%, taking into account the speed variations in the link among the servers, the percentage obtained proved to be a good result, making viable the use of this approach in a continuous manner.

7. CONCLUSIONS AND FUTURE WORK

It was possible to demonstrate, from the case study of integration with a taxi monitoring application, that the proposal of i9ITS architecture, using the SOA approach, promotes effectively and efficiently the retaining of information related to geo-referenced data of urban mobility.

Thus, it is concluded that the proposed architecture meets the performance requirements and efficiency of this case study, as well to other possibilities aiming to meet the demands and challenges related to ITS.

Finally, there are several possibilities for future work from this study. One of these possibilities would be to evolve this study in order to promote the retention of information from other domains related to smart cities (such as Health, Education, Environment, Security, Energy Efficiency, among others) for the provision of open data.

Other work would be to define the construction of Ontology

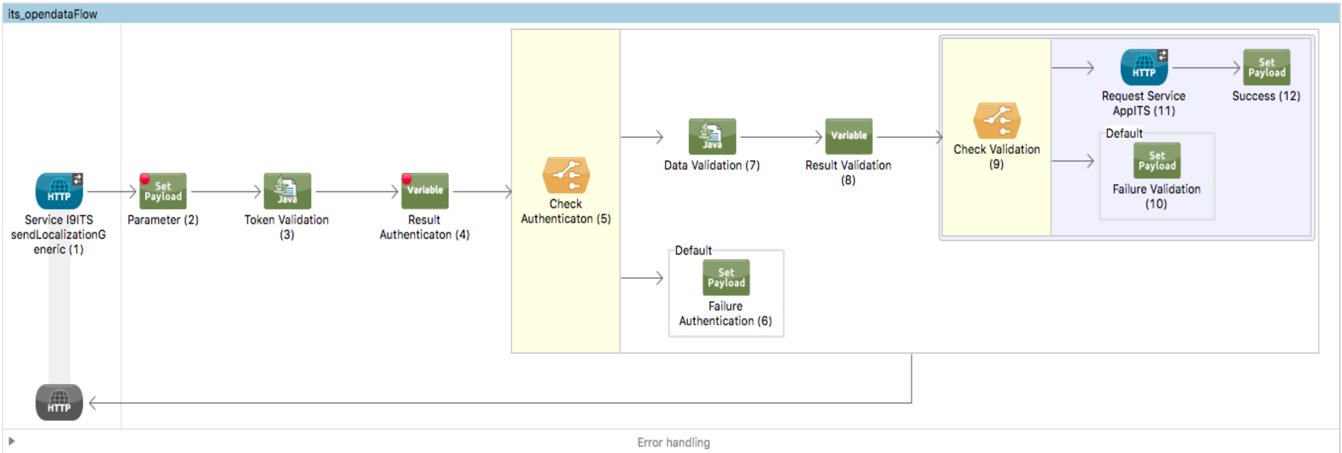

Figure 5: Execution flow in Mule Standalone ESB (Source: Authors).

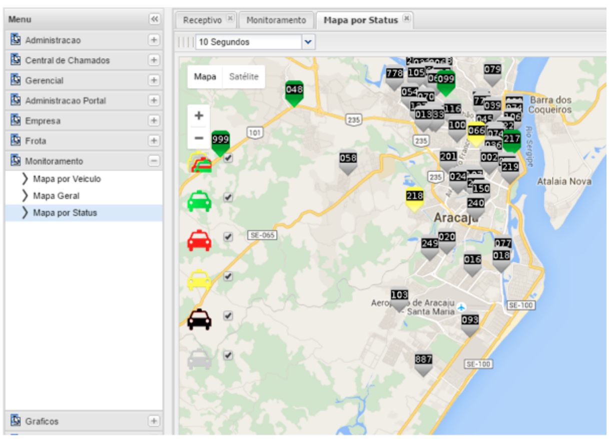

Figure 6: Screenshot of the cab monitoring system (Source: Authors)

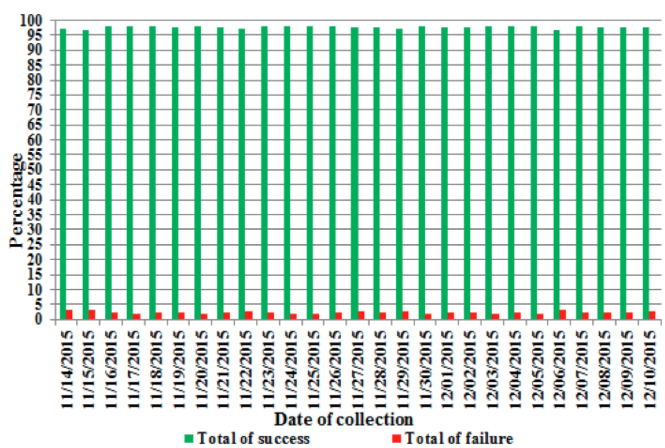

Figure 7: Rate of Success and Failure request that occurred between the Georeferencing Application and i9ITS architecture (Source: Authors).

Table 2: Data Collection Performed to the analysis of the Efficiency of i9ITS Architecture Applied in the Case Study.

Date of collection	Total of requests	Total of success	Total of failure	Average time of requests (ms)
11/14/2015	62.500	60.625	1.875	124
11/15/2015	8.789	8.492	297	123
11/16/2015	245.312	240.127	5.185	122
11/17/2015	249.277	244.122	5.155	122
11/18/2015	247.233	241.909	5.324	121
11/19/2015	313.967	306.769	7.198	122
11/20/2015	285.053	279.563	5.490	121
11/21/2015	197.671	193.035	4.636	122
11/22/2015	35.197	34.221	976	123
11/23/2015	285.511	279.350	6.161	122
11/24/2015	345.931	339.160	6.771	121
11/25/2015	228.171	223.931	4.240	122
11/26/2015	279.380	273.402	5.978	121
11/27/2015	299.175	291.428	7.747	122
11/28/2015	257.847	251.808	6.039	122
11/29/2015	75.369	73.356	2.013	122
11/30/2015	311.443	305.434	6.009	121
12/01/2015	329.675	321.745	7.930	121
12/02/2015	364.628	356.576	8.052	121
12/03/2015	369.806	362.242	7.564	121
12/04/2015	319.518	312.759	6.759	121
12/05/2015	318.054	311.564	6.490	121
12/06/2015	24.522	23.766	756	122
12/07/2015	321.641	314.638	7.003	121
12/08/2015	80.731	78.747	1.984	122
12/09/2015	291.658	284.616	7.042	121
12/10/2015	307.172	299.266	7.906	122

and Taxonomy to categorize and list the services available on this architecture and so standardize the availability of information from them via ESB and service portals. It is also possible to build algorithms to detect the reliability of the data retained to verify the authenticity and quality, defining if they would be considered or not.

Studies to the use of mining algorithms to solve problems related to urban mobility such as smart parking Systems through analyzing the geographical positions identifying available spaces; traffic management system, allowing to predict possible areas of congestion; the public transportation system efficiency analysis, allowing a better distribution of the fleet on the routes.

8. ADDITIONAL AUTHORS

Additional authors: Frederico A. S. Gentil (Faculdade de Administração de Negócios de Sergipe -FANESE, email: fredgentil@gmail.com)

9. REFERENCES

[1] C. H. R. d. Carvalho, B. A. Furtado, B. d. O. Cruz, R. Pereira, and M. d. P. Morais. Mobilidade urbana e posse de veículos: análise da pnad 2009. 2010.

[2] O. Definition. The open definition. http://opendefinition.org/. Accessado: 11/11/2015.

[3] H. B. Diniz, E. C. Silva, and K. S. da Gama. Uma arquitetura de referência para plataforma de crowdsensing em smart cities. 2015.

[4] Esri. Esri's solutions for geospatial soa. http://www.esri.com/news/arcnews/summer06articles/spatially-enabling.html. Accessado: 18/02/2016.

[5] O. K. Foundation. Open data handbook. http://opendatahandbook.org/guide/en/. Accessado: 18/11/2015.

[6] E. Froelian and G. Sandhaus. Conception of implementing a service oriented architecture (soa) in a legacy environment. Technical report, ild Schriftenreihe Logistikforschung, 2012.

[7] L. F. Herrera-Quintero, F. Maciá-Pérez, D. Marcos-Jorquera, and V. Gilart-Iglesias. Wireless sensor networks and service-oriented architecture, as suitable approaches to be applied into its. In *Telematics and Information Systems (EATIS), 2012 6th Euro American Conference on*, pages 1–8. IEEE, 2012.

[8] S. Isotani and I. I. Bittencourt. *Dados Abertos Conectados*. Novatec Editora, 2015.

[9] L. James. Defining open data. http://blog.okfn.org/2013/10/03/defining-open-data/. Accessado: 11/02/2016.

[10] K. Kuikkaniemi, G. Jacucci, M. Turpeinen, E. Hoggan, and J. Müller. From space to stage: How interactive screens will change urban life. *Computer*, (6):40–47, 2011.

[11] C. Li. Travel information service system for public travel based on soa. In *Service Operations and Logistics and Informatics (SOLI), 2010 IEEE International Conference on*, pages 321–324. IEEE, 2010.

[12] X. Lu, Y. Zou, F. Xiong, J. Lin, and L. Zha. Icomc: invocation complexity of multi-language clients for classified web services and its impact on large scale soa applications. In *Parallel and Distributed Computing, Applications and Technologies, 2009 International Conference on*, pages 186–194. IEEE, 2009.

[13] R. Nasim and A. Kassler. Distributed architectures for intelligent transport systems: A survey. In *Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on*, pages 130–136. IEEE, 2012.

[14] S. C. A. of Governments. Architecture elements. http://www.scag.ca.gov/programs/Pages/ArchitectureElements.aspx. Accessado: 11/02/2016.

[15] A. Ojo, E. Curry, and F. A. Zeleti. A tale of open data innovations in five smart cities. In *System Sciences (HICSS), 2015 48th Hawaii International Conference on*, pages 2326–2335. IEEE, 2015.

[16] M. Shoab, K. Jain, M. Anulhaq, and M. Shashi. Development and implementation of nmea interpreter for real time gps data logging. In *Advance Computing Conference (IACC), 2013 IEEE 3rd International*, pages 143–146. IEEE, 2013.

[17] Z. Xiaorong. Cross-domain authentication model in soa based on enterprise service bus. In *Computer Engineering and Technology (ICCET), 2010 2nd International Conference on*, volume 5, pages V5–78. IEEE, 2010.

[18] T. Zhu and Z. Liu. Intelligent transport systems in china: Past, present and future. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2015 Seventh International Conference on*, pages 581–584. IEEE, 2015.