An Adaptable Approach for Indoor Location

Mário Andrade Vieria de Melo Neto
Federal Institute of Rio Grande
do Norte – Natal, Brazil
Email: mario.melo@ifrn.edu.br

Gibeon Soares de Aquino Júnior
Federal University of Rio Grande
do Norte – Natal, Brazil
Email: gibeon@dimap.ufrn.br

***Abstract*—In general people often spend 80-90% of their time in indoor environments, which include shopping malls, libraries, airports, universities, schools, offices, factories, hospitals, among others. In these environments, GPS does not work properly, causing inaccurate positioning. Currently, when performing the location of people or objects in indoor environments, no single technology can reproduce the same results achieved by the GPS for outdoor environments. One of the main reasons for this is the high complexity of indoor environments where, unlike outdoor spaces, there is a series of obstacles such as walls, equipment and even people. Due to this, it is necessary to consider the use of information from multiple sources using different technologies. Thus, this work proposes an adaptable approach for indoor location, which allows the use and combination of different technologies, techniques and methods in this context.**

I. INTRODUCTION

Nowadays, it is clear that location systems are increasingly present in people's lives. These systems can help people solve different kinds of problems in a great variety of situations. People in general often spend 80-90% of their time in indoor environments [10], which include shopping malls, libraries, airports, universities, schools, offices, factories, hospitals, among others. Because of this, services that allow location in indoor environments have been gaining special attention. One of the reasons for the increase in popularity for this type of application is the popularization of portable devices such as cell phones, smartphones, PDAs, tablets and notebooks, which already have many built-in hardware such as WiFi, Bluetooth, GPS and inertial sensors.

There has been an increase in the demand for the location of people or objects in indoor environments to be a reliable and accurate [4]. In this sense, large technology companies such as Google and Apple are currently investing in this research area in order to develop solutions for location in indoor environments. This shows that this problem remains unsolved, that is, there is still no technology or combination of technologies that can solve the problem in an acceptable manner and with low costs [5]. One of the main reasons for this is the high complexity of indoor environments where, unlike outdoor environments, there are differents obstacles such as walls, equipment and even people [7].

Thus, it is necessary that the solutions proposed to solve the problem of location in indoor environments take into account the complexity of these environments. Melo and Aquino [6] showed that there is a tendency for these solutions to combine the use of different technologies, sources of information, location techniques, among other features, which will allow the adaptation of the solutions to the various complexities that can be found in these environments. In order to come up with a solution that fits all of the environments and that can be adapted to their specific characteristics, this paper proposes an adaptable platform that enables the combination and use of many techniques and technologies in order to obtain the location of people or objects in indoor environments.

The paper is organized as follows: Section II will be dedicated to the presentation of the proposed platform, detailing its requirements, architecture and components. In Section III, the evalutation of the platform's behavior applied to the context of location in indoor environments using WIFI and RFID technologies is presented. Section IV presents some related works, comparing them and pointing out strengths and weaknesses. Finally, Section V presents the final considerations, as well as future studies for this work.

II. PLATFORM ARCHITECTURE

The proposed platform defines a number of components that may have their functionalities adapted in order to enable the reception, processing and storage of many heterogeneous data that make up an indoor environment. Furthermore, the possibility of the adaptation of its components allows the platform to be adapted to different indoor environments and enables it to meet the specificities found in each one of them. Hence, new approaches, algorithms or techniques can be added without the need to adjust the base architecture.

In Figure 1, the general architecture of the platform is presented. This platform has a predefined set of components which are called main components. The proprosed plataform has 8 main components which are: I/O Manager, Request Manager, Map Manager, Data Publication Manager, Location Manager, Event Manger, Things Manager and Data Storage Manager. Each one of these components provides an adapter interface (AI) so that they may have its functionalities adapted by the adapter components. Moreover, given the separation of the architecture in different components, it is possible to adapt specific components, which making it optional to extend all of the platform's components in order to include or modify the capabilities.

The process of adapting the functionalities of the main components are on each component allows the adaptation of its functionalities through the AI using the Whiteboard Pattern [3]. Instead of searching in a directory for the component that

DOI reference number: 10.18293/SEKE2016-130

implements the required functionality, the main component registers the interest in components that are able to implement the adaptation of its functionalities. So whenever an adapter component is registered in the component directory, it checks if there is any main component interested in it. If there is, it is notified. Upon receiving the notification, the main component has access to the registred component instance, which performs the adaptation of its functionality.

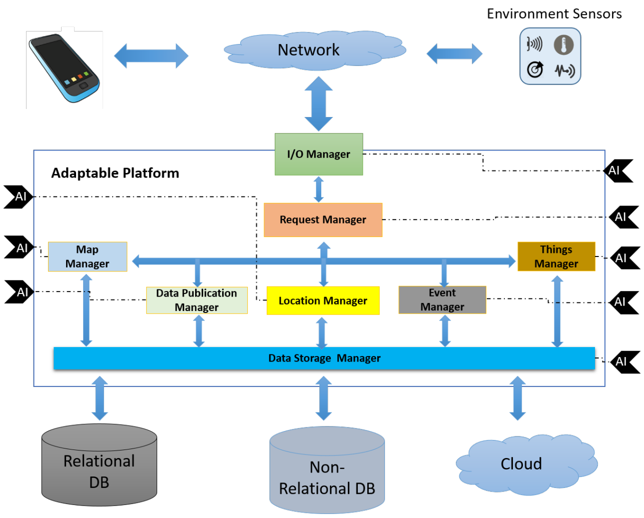

Fig. 1: Adaptable Platform Architecture Overview

A. *I/O Manager*

This is the component that handles the input and output towards the platform. The main goal of this component is to ensure that the platform enables the handling of heterogeneous data, allowing the input of any information regardless the technology or unit used in order to provide many types of location services.

B. *Request Manager*

This component acts as a dispatcher identifying the type of request sent to the platform, forwarding the request to the main component so it can perform its processing. This component can have its functionalities adapted. Thus, new requests for processing flow can be created, considering aspects not implemented by the main components.

C. *Data Publication Manager*

It is the component responsible for allowing the data produced on the platform to be accessed by client applications. This component will carry out the provision of information received and produced by the platform using the content providers made available by the *Data Storage Manager* component in order to provide this information.

D. *Map Manager*

If the request type is related to the maps domain, this is the component that will be responsible for receiving the request, transforming the information for this domain and invoking the service that will process the request.

E. *Location Manager*

This is the main component of the platform and is responsible for performing the processing of location information. This component's extensions must be able to execute three different behaviors. The first behavior is to turn the received information into data that can be processed by the component. The second identify the best strategy for processing this information. In this processing stage, it might to use different location algorithms, a combination of them as well as single algorithm that can perform the fusion of information in order to generate new types of information. Finally, the received and processed information must be sent to storage so that other functionalities can use them.

F. *Event Manager*

This component will allow the use of the Push interaction mode [1] using the , in which it is possible to perform the registration of events by the implementation of the publish-subscribe pattern.

G. *Things Manager*

This component will be responsible for managing the platform, performing tasks such as registration, modification and removal of "things" that are necessary to platform operation. These "things" can be environments' equipment, devices, management data, among others.

H. *Data Storage Manager*

This component will serve as a data storage provider to the main components access the information persisted by the plataform. This component support different forms of data storage, i.e., relational database, non-relational database, cloud storage services and others.

III. Evaluation

In order to evaluate the proposed platform, an Android application was used to collect data from WIFI access points present in the given environment. In addition to these data, other data from RFID tags found in specific points of the environment were also collected. In every data collection, the data was sent to the platform, which processe it and calculate the estimated position of the device. In order to carry out this evaluation, the scenario used to for this is presented in the Figure 2. To execute this evaluation it was needed to create four adapter componentes which are: **RFID Service** – this component implements the functionality that processes data obtained from the RFID tags by the client applications, **WIFI Fusion Service** – performs the adaptation of the location functionality obtaining the user location using WIFI and RFID data, **WIFI Location Storage and RFID Location Storage** – These components are responsible for performing the storage of the WIFI and RFID data received by the platform.

The environment used for this assessment is a conventional office in a commercial building measuring 115 m^2. In this environment, there are four access points, so the existing infrastructure was used to perform this assessment. In addition,

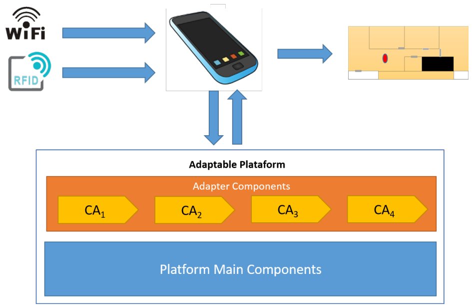

Fig. 2: Evaluation Scenario

there is one RFID tag attached to a wall in one of the rooms in the environment. Thus, a walking test was performed within the office and its result is shown in Figure 3. In the Figure, the environment map shows two routes. The dashed line represents the actual route, or the route that was taken by the user. The solid line represents the route calculated by the platform using the strategies defined by the adaptable components.

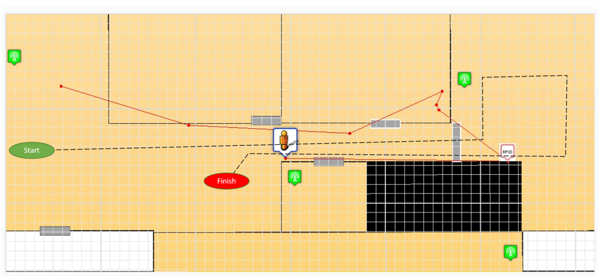

Fig. 3: Android app using the proprosed platform for indoor location

The evaluation was performed under two facets: main components processing time compared to the total plataform processing time and the easiness of the creation and incorporation of adapters components. In order to assess the processing time of the designed platform, the measurement of the time required for a request to be fully processed within it was performed. Thus, the time spent from the moment the request was received until the response was sent to the requestor was measured. The requests send to the platform was to calculate the indoor location of the device. This request was chosen because is the most costly to the platform. The execution context in which the platform was implemented is an Intel Core i7 1.8 GHz machine with 8 GB of ram. However, for this assessment, the memory usage was limited to 512 MB. Accordingly, the average time that it takes a certain amount of requests sent at a time to be processed by the platform was calculated. The results are shown in Figure 4, in which the X axis represents the amount of requests sent and the Y axis represents the average processing time in milliseconds. In addition, for each amount of requests sent, ten samples were collected and their average time was calculated.

It is possible to note that the main component processing time does not exceed 0,11 milliseconds which represents

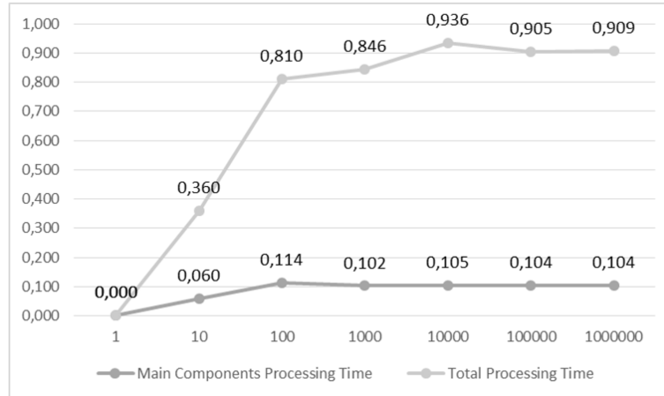

Fig. 4: Requests Processing Time

11% of the total time to processing the requests sent to the platform. This demonstrates that the greatest amount of average processing time is spent with the logical associated with the adapters components. Another fact that was possible to note is that the plataform was able to keep the average processing time in stable values even increasing load 100 times higher.

TABLE I: NUMBER OF LINES OF CODE FOR THE EVALUATION

Component	Lines of code	Lines for adaptations
WIFI Fusion Service	157	16
WIFI Location Storage	127	9
RFID Service	160	16
RFID Location Storage	143	9
Total	587	50

Regarding the easiness to create and incoporate adaptations, we collected the amount of lines of code implemented in each of the adapter component, as shown in Table I. On this count, all of the lines of code in the source code's file were accounted for, including imports, statements, etc. In addition, we also counted the lines of code that are directly related to tasks that is necessary to incorporate these adapter components to the platform. Therefore, were required total 64 lines of code for the components to be coupled to the platform. Thus, it was necessary less than 9% of total lines of code to incorporate the adaptabilities the platform.

IV. Related Work

In this section, related works directly associated with what was proposed in this work will be presented. We will highlight the main contributions of each of the listed works and will try to show the main differences between them and the proposed approach.

The Location Stack Architectural model introduced by Hightower et al. [2] shows a standardized model for developing location systems using a layer architectural model which refers to standard ways to perform the combination of data from different sources using a software architecture based on the Open System Interconnect (OSI) network communication model. As the main difference between the Location Stack and the proposed platform's architecture, it is possible to note that the fact that it defines rigid layers - just like the OSI model in which a particular layer only receives data from the layer right below it and only exports the processed data to the layer right above it - makes the architecture only a little flexible. This little flexibility makes it hard to extend the functionalities, which differs greatly from the proposed architecture.

In Najib et al. [8], a middleware for location in indoor environments called MapUme was presented. The MapUme allows the fusion of data from various types of sensors. The architecture used is based in an architecture proposed by Hightower et al., in which it is organized in layers. Also, it uses the service-oriented architectural model. Despite being based on a modular architecture and divided into layers, this middleware allows the adaptability of its components. However, the adaptability is only allowed for some of the architecture's components. Even when the adaptability is allowed, it must occur with a high degree of coupling, since the new implementation must be built and coupled to the middleware. This feature makes it very different from the proposed platform, which allows the inclusion of new features without the need for interrupting the platform implementation.

In Ranganathan et al. [9], a middleware sensitive to location called "MiddleWhere" is proposed. This platform integrates multiple location technologies as well as presents the consolidated location data to customer applications. In order to do this, it uses a layered architecture to collect information from sensors and persist the information in a special database; an intelligence engine to perform data fusion in order to obtain the location information; and a service layer to make such information available. It also allows the pull and push interaction mode for communication with applications. Unlike the approach proposed in this paper, this middleware only allows a small degree of adaptability, making it only possible to vary the runtime of the insertion and removal of new sources of information. The proposed platform's architecture allows the use of new and different sources of information at runtime, as well as allows the modification of any of the behaviors found in the platform.

V. Conclusion and Future Works

This work presented the design, implementation and assessment of an adaptable platform for location of people and objects in indoor environments, which allows the use of different technologies, information sources, techniques, among other characteristics, in order to adapt itself to different and complex indoor environments. Furthermore, it was shown that the proposed platform is divided into main and adapter components. These main components are provided by the platform's architecture and can have their functionalities adapted by the adapter components.

The proposed platform was evaluated using an implementation in which WIFI data obtained from access points and RFID data obtained using tags were used. To collect such data an Android application was built. Furthermore, the platform were evaluated under two aspects: main components processing time compared to the total plataform processing time and the easiness of the creation and incorporation of adapters components. The main components processing time cause minor impact on the total process time, even with increasing amounts of requests sent. Regarding the easiness of creating and incorporating adaptability's, indicates to be an easy task. We realized that the amount of necessary lines of code on the specific parts of the platform up less than 10% of the total.

As future works, we intend to assess platform operation using other technologies and techniques in order to identify potential improvements in the proposed architecture. Furthermore, there is the intent to evaluate other characteristics related to platform implementation, such as processing and distributed scalability. Besides, we also intend to assess the use of cloud computing concepts and big data.

VI. Acknowledgment

This work was partially supported by the National Institute of Science and Technology for Software Engineering (INES) funded by CNPq under grant 573964/2008–4

References

[1] Manish Bhide, Pavan Deolasee, Amol Katkar, Ankur Panchbudhe, Krithi Ramamritham, and Prashant Shenoy. Adaptive push-pull: Disseminating dynamic web data. *Computers, IEEE Transactions on*, 51(6):652–668, 2002.

[2] Jeffrey Hightower, B. Brumitt, and G. Borriello. The location stack: a layered model for location in ubiquitous computing. In *Mobile Computing Systems and Applications, 2002. Proceedings Fourth IEEE Workshop on*, pages 22–28, 2002.

[3] Peter Kriens and B Hargrave. Listeners considered harmful: The whiteboard pattern. *Technical whitepaper, OSGi Alliance*, 2004.

[4] F. Lemic, V. Handziski, N. Wirstrom, T. Van Haute, E. De Poorter, T. Voigt, and A. Wolisz. Web-based platform for evaluation of rf-based indoor localization algorithms. In *Communication Workshop (ICCW), 2015 IEEE International Conference on*, pages 834–840, 2015.

[5] Dimitrios Lymberopoulos, Domenico Giustiniano, Vincent Lenders, Maurizio Rea, Andreas Marcaletti, et al. A realistic evaluation and comparison of indoor location technologies: Experiences and lessons learned. *ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 178–189, 2015.

[6] Mário Melo and Gibeon Aquino. Categorization of technologies used for fingerprint-based indoor localization. In *Systems and Networks Communications, 2015. ICSNC'15. Eleventh International Conference on*, pages 25–29. IARIA, 2015.

[7] Mário Melo and Gibeon Aquino. A taxonomy of technologies for fingerprint-based indoor localization. In *7o Simpósio Brasileiro de Computação Ubíqua e Pervasiva (SBCUP 2015)*, pages 111–120. SBC, 2015.

[8] W. Najib, M. Klepal, and S.B. Wibowo. Mapume: Scalable middleware for location aware computing applications. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–6, Sept 2011.

[9] Anand Ranganathan, Jalal Al-Muhtadi, Shiva Chetan, Roy Campbell, and M Dennis Mickunas. Middlewhere: a middleware for location awareness in ubiquitous computing applications. In *Proceedings of the 5th ACM/IFIP/USENIX international conference on Middleware*, pages 397–416. Springer-Verlag New York, Inc., 2004.

[10] M. Simoni, M.S. Jaakkola, L. Carrozzi, S. Baldacci, F. Di Pede, and G. Viegi. Indoor air pollution and respiratory health in the elderly. *European Respiratory Journal*, 21(40 suppl):15s–20s, 2003.