A Personalization System for Data Visualization Platforms

Orlando Belo[1], Paulo Rodrigues[1], Rui Barros[1]
[1] University of Minho
Portugal
obelo@di.uminho.pt

Helena Correia[2]
[2] Portugal Telecom Inovação e Sistemas, SA
Portugal

ABSTRACT: *Today dashboards are a clear factor of differentiation for decision-makers, providing critical business indicators in useful time. They are very important data visualization mechanisms, simple to interpret, easy to deal, and fast on showing pertinent data. Dashboards provide means for reducing time of information analysis, disposing business indicators in quite understandable and attractive interface platforms – not the ability to evolve, to adapt by themselves to new user needs, tendencies or preferences. In order to provide such abilities, we designed and implemented a personalization system for data visualization platforms. All system's services, ranging from data gathering to data profiling, were developed using a community of autonomous agents with the ability to act together using a private data cloud as a cooperative environment. In this paper we present how the system was conceived, its functional architecture and services, and how agents were implemented, especially the ones related to the personalization of business dashboards.*

Keywords: Data Visualization, Multi-agent Systems, On-Line Analytical Processing, Usage Profiling, Adaptive Dashboards, and JADE

Received: 28 July 2016, Revised 29 August 2016, Accepted 7 September 2016

1. Introduction

For any decision maker, having the right information at the right time is so precious that its value cannot be determined exactly. For a long time, companies' managers, and decision makers in general, sought to create and maintain information systems that enable them to get quickly and easily the most relevant information in the field of their business activities. However, such systems are not always be prepared to satisfy such goals. Sometimes, a lot of cases raise serious difficulties in the process of gathering and consolidating the information decision makers need, changeling the generation of a single and consolidated view

and the creation of homogeneous business scenarios. In such cases, analytical systems assume a prominent position within companies that are concerned about the development of their business processes, and in particular with direct competitors in their market. As we know, these systems allow for dealing with the information in a more accurate way, supporting decision-making activities in a very effective way.

Today, the most rudimentary analytical system offers a range of services and presentation formats so extensive and versatile that the majority of analysis models are easily implemented and customized accordingly to the requirements of decision agents. This ensures to get a consolidated view of corporate business models as well as allows for navigating along the several dimensional perspectives of analysis [1]. By nature, data navigation processes are quite easy to sustain in analytical systems due to their data storage structures characteristics – hyper-cubes –, which accommodate selective information that can be explored in a very versatile way using a group of special operators – e.g., roll-up, drill-down, pivoting, etc. Hyper-cubes sustain business oriented multidimensional views of data defined and maintained according to different perspectives of analysis. However, despite their sophistication and power, these structures and operators alone are not enough to ensure all the requirements of analysis that decision makers have.

Usually, decision-makers are users who require simple but sophisticated operational means, easy to learn and easily to configure according to their usage. In practice, they require analytic data exploration means with intuitive interfaces, with which they do not need to spend a lot of time performing secondary tasks, such as configuration analysis or presentation formats, just to name a few. They want to focus on what really matters: the analysis of information and the decision-making process. One of the tools that more evolved in this direction in recent years were dashboard systems. Today, dashboards have great popularity in the field of decision support systems. Their simplicity and clarity in presenting and communicating relevant information have made them a very powerful information communication tool. Nevertheless, dashboards are not yet explored in its full potential [2]. To create a quality dashboard equal to or higher than the expected, it is necessary to understand how to define and organize it graphically, as well as to know the context of application and preferences of its users. However, it is their ability to reflect user preferences that makes a dashboard having more impact on the everyday lives of decision-makers.

Dashboard adaptation to usage profiles was already subject of some research and development work in the field of analytical systems. To be successful this data visualization customization is only possible if the system has prior knowledge about users' needs and preferences, i.e., if it has the ability to make some kind of usage profiling. There are several ways to do that. We can use some usage questionnaires [3], usage feedback by asking user to rate the usefulness of the system tool [4], or usage profiling by capturing and recording all actions the user had done so far in the system. With the latter technique, it is possible to determine very concrete usage patterns, which allow for knowing the features users most use, the time they spend in particular area of the system, or which queries they use to post. All of these can be done without being necessary to ask the user.

After a brief period of observation and requirements analysis on some decision support systems environments, we felt that dashboards could be more practical and effective if they have some kind of ability to evolve accordingly the usage decision makers made of them. This would be a clear advantage aspect, a real differentiation factor, especially if dashboards will have mechanisms for acquiring and reflecting user preferences [5] [6]. However, in a typical corporat environment, decision-makers use to get the data they need in a very diverse group of information sources or alternatively on an analytical system, if available. Any of these data gathering processes is difficult and expensive, requiring very demanding computational platforms in part due to the great volume of ad hoc filtering and aggregation tasks involved with. In this work we were especially interested in the second modality of data gathering. Our main concern was to distribute every single task, ranging from data gathering to profiling, by high sophisticated entities, and putting them working together in order to support an effective dashboard system with adaptive abilities. The implementation of a multi-agent system [7-10] perfectly satisfies all these requirements for the implementation of a sophisticated analytical system, providing means and characteristics that fit quite well in a real world analytical scenario, where we face frequently a real distributed corporate data warehousing system.

In this paper we present a multi-agent system that was designed and implemented to support a personalization system for data visualization platforms. We will discuss how agents were implemented, as well as the environment they share, and services they provide, especially the ones related to the personalization of decision-makers' dashboards. The paper is structured as follows. Section 2 presents some research work related to this project. Next, in section 3, we describe how we designed the personalization system as a multi-agent system, the agents involved with, the structures they use, and how they work on a specific dashboard personalization process. Finally, in section 4, we present some conclusions and a few guidelines for future work.

2. Related Work

The use of visual means to represent information is a practice that has been present for hundreds of years in several areas of activity, ranging from academic to business [11]. It was in the late 18th century, early 19th century, that William Playfair has created some of the diagrams that are commonly used today in the representation of information – e.g., bar graphs, line graphs and pie charts. Despite all this, his work had no major impact while alive [12]. Currently William Playfair is known for his huge contribution for the development of graphic techniques - for many he is the "father" of data visualization. In the following years other types of graphical representations appeared. However, it was John Snow who first demonstrated the colossal importance of using visual representations of information to create one of the first data visualization maps with the goal to determine the cause of the cholera epidemic of 1854 [13]. Some years later, a nurse and statistics, Florence Nightingale, used some pie charts to show that the leading cause of death of the British soldiers during the Crimean War (1853-1856) was the lack of hygiene in hospitals that treat wounded in combat [14]. Later, in 1861, Charles Minard developed a new chart, mapping the tragic march of Napoleon to Moscow [15], which was rated by Edward Tufte, in 1983, as possibly being the best statistical graphic ever. Many other works related to the visual representation of information appeared next. However, it was only in the 20th century that professor John Tukey, in 1977, from the Princeton University, showed the true power of using data visualization techniques as a mean to explore and represent data [16]. But among all the works presented till then, one appeared with a great impact, the work presented by Edward Tufte in 2001 [17].

More recently, data visualization tools have played a very important role in the field of Business Intelligence and Business Analytics, which led to the emergence of numerous specialized solutions for data visualization. However, despite these tools being extremely powerful, useful, and attractive, in fact they reveal some disadvantages, because often they require specialized knowledge to be used, and do not reflect in their services users' preferences or tendencies. Today, dashboards stand out as one of the most efficient and popular data visualization technique. The ability they present to view various perspectives of analysis over several information systems opened the door to numerous opportunities in the business world, in terms of analysis and monitoring of a company [16].

One way to make data visualization mechanisms more sophisticated is using personalization techniques [18] [19]. From long time personalization was used to receive users preferences and improve system's behavior in general. There are many reasons that can be used to justify personalization, but all of them aim to monetize the resources involved in business activities targeting a higher return of investments and a better quality of service. However, in analytical systems it is not common to find solutions incorporating personalization strategies, despite having many solutions in other application areas. For example, the application of data mining techniques in customizing websites [20], the customization of print data in e-commerce environments [21], or the use of customization techniques based on search contexts systems [22], are some good examples that reveal the importance of personalization in real-world applications. Additionally, in some other cases of personalization, we found software agents assisting users, managing personal information spaces [23], customizing information for reusing models for information system engineering [24], or personalizing Web services [25]. Today, personalization is a key aspect on every application, as a way to provide the means to tailor users' needs over time on sophisticated interface platforms.

3. Personalizing Data Visualization Processes

3.1 General Overview

Usually, the activities developed by decision support systems are supported by a wide variety of data. This is required by the nature of the various analytical perspectives of decision makers involved with. But not always the amount of data is an effective synonymous of useful information, particularly if it cannot be transformed into business knowledge for helping decision makers. Business dashboards are especially designed to provide to decision makers the information they need, providing a simple an intuitive way to manipulate it autonomously and independently. To turn this possible, the most sophisticated data visualization techniques are used to ensure expeditious means and mechanisms for handling such large amounts of data. Such mechanisms are indeed quite important and help decision makers to do their jobs as expected. However, analytical platforms having data visualization static configurations end to be unattractive to users, which expect a more intelligent behavior of the system they are using. The inability to adapt to new functional requirements or to reflect the preferences of its users, for instance, may be one of the strongest reasons that often leads to abandon some of analytical platforms. Decision-makers like to be surprised with new data visualization proposals – e.g., dashboards, key performance indicators, or ordinary reports –, which reveal new data (some times also with other visualization formats) prepared according to the information manipulated on previous decision-making processes of past analytical sessions.

The introduction of adaptive mechanisms is something that can greatly assists a data visualization process personalization, automating data exploration processes, reducing the need to perform repetitive tasks (the most predictable tasks), and increasing the efficiency of using the system. Thus, performing a secondary task such as configuring a dashboard, setting data formats or filtering preferences, or defining a time period for analysis can be performed automatically by the system, giving more time to users to refine and improve their analysis and decision making tasks. In order to provide such abilities, we modified a little bit the conventional design of an analytical system, placing a new functional component between its platforms for data interfacing and data processing (Fig. 1). This new component was designed in order to capture all forms of interaction between users (decision makers) and the system itself. When a user requests to view a given set of data using a dashboard, for instance, this new component collects a copy of the multidimensional query (MDX) produced by the interface platform, and saves it into its local database system for further analysis. Later, this component will analyze all the queries it collected, analyzes them and identifies pertinent data usage,such as the most requested data, the periods of greatest activity, the sequences in which the queries were posted, or the querying preferences of the users. With this, it constructs or updates analytical user profiles.

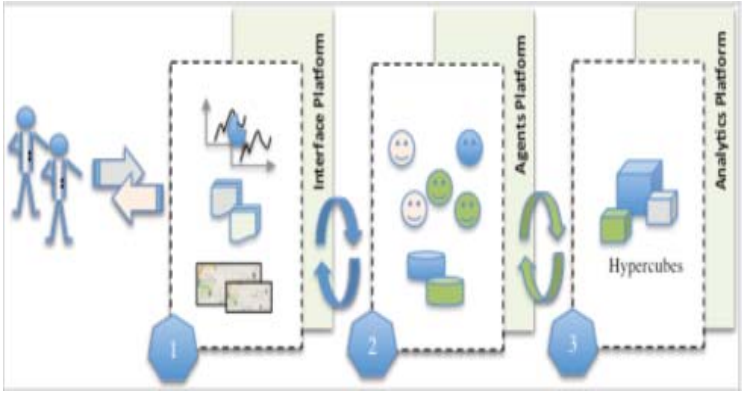

Figure 1. System's Functional Architecture

This intermediary platform acts as a typical middleware having the ability to "rethink" data visualization structures, based on usage profiles. So, every time it concludes that a user (or a group of users) requires systematically a package of data in the same period of time (just say Friday morning), using a specific dashboard or a data grid, the system will rearrange automatically the correspondent data visualization mechanisms (metadata), launches the queries involved with to collect data in the analytics platform, and shows the results in the moment that the period occurs again – is just like an a priori data personalization process. To produce a component like this is a complex task and difficult to achieve. It demands a modular and incremental development approach to receive gradually new functionalities and services that make possible to reflect the evolution of the trends and preference of users, and ensure to handle all interactions between different systems' components. As we know, they change a lot over time. What is relevant or important today for a user, provably tomorrow will be not. This is quite frequent to happen in analytical systems when supporting decision-making activities in very competitive markets. Taking this into consideration, we decided to design and implement this component as a multiagent system [9] [10] [26], in which agents were categorized and divided accordingly the different major tasks involved with personalizing an analytical system, namely data visualization, profiling evaluation and personalization, and multidimensional data access.

3.2 System's Agents

The advantages of using agents for personalizing data visualization mechanisms are various. Among them, we can refer the possibility of modeling the adaptive data visualization system through the definition of areas of competence under the jurisdiction of a particular group of agents that have the ability to interact with each other, solving problems together on jointly cooperation processes. All these characteristics improve system's functionalities as well as provide for a more robust and scalable system. Additionally, with agents we can have a better distribution of the system's computational resources, and improve system performance through the provision of a set of distributed processing entities more robust, reliable, maintainable, and reusable.

System's agents are specialized entities with specific skills and knowledge that define their competence accordingly the role they have in the system. Essentially, they were organized into three distinct classes accordingly the major tasks of the

system. The first class integrates the data providers, which are the agents responsible to access multidimensional data objects in the system's private cloud, for collecting the data to satisfy users' queries launched through the system's interfaces platforms. They also store the data objects locally in a multidimensional database supported by a Mondrian server, keeping them alive as far as analytical sessions remain active. The system allows for having several instances of these agents running simultaneously. But this rarely happens, because these agents may attend at the same time several analytical sessions answering and satisfying all sessions' queries. So, we need only a single instance of these agents running on the system. Despite having access to all the data available in the system's cloud, these agents only collect the data to satisfy the needs of system's users. A private data cloud was implemented to support a global multidimensional database repository, ensuring a natural distribution of information sources that occurs in a conventional analytical system, providing a right resource allocation in cases when the system scales, and, of course, to sustain a conveniently mean to agents access and share analytical data.

The data visualizers constitute the second class of agents. They are responsible to support users on system's interface platforms providing the structure (metadata) and the data necessary to sustain every data visualization structures that are active in the interface platforms, especially the ones related to system's dashboard. These agents feed the interface platforms with data sent by a data provider that answered to queries posted by the users of such platforms. However data visualizers never know the queries that were previously sent to a data provider by an interface platform. They only know the interface platform they are working on through a message sent by a data provider jointly with the queries' results. Additionally, data visualizers are responsible to monitor and communicate errors that occur on the system's dashboards.

Finally, the third class integrates the dashboard restructurers. These are the most important system's agents, since they are responsible to personalize the data visualization structures of the system's interfaces platforms accordingly the usage profiles of the users. They act based on a predefined agenda and define the time intervals where user preferences detection and data visualization configurations predictions algorithms will be activated. The behavior of these agents is set at the time of their creation in the system, independently of the class to which it belongs. Their tasks are defined through a specific setup configuration file containing the location of their data gathering agendas and the configuration of the databases they use – Fig. 2 presents an example of a configuration file of a data provider.

```
<?xml version="1.0" encoding="UTF-8"?>
<tasks>
  <task>
      <queries>
        <query tableName="account">SELECT * FROM account WHERE ? </query>
        <query tableName="category">SELECT * FROM category WHERE ? </query>
        <query tableName="currency">SELECT * FROM currency WHERE ? </query>
        <query tableName="customer">SELECT * FROM customer WHERE ? </query>
        <query tableName="days">SELECT * FROM days WHERE ? </query>
        <query tableName="department">SELECT * FROM department WHERE ? </query>
        <query tableName="position">SELECT * FROM position WHERE ? </query>
        <query tableName="product">SELECT * FROM product WHERE ? </query>
        <query tableName="product_class">SELECT * FROM product_class WHERE ? </query>
        <query tableName="promotion">SELECT * FROM promotion WHERE ? </query>
        <query tableName="region">SELECT * FROM region WHERE ? </query>
        <query tableName="reserve_employee">SELECT * FROM reserve_employee WHERE ? </query>
        <query tableName="sales_fact_1997">SELECT * FROM sales_fact_1997 WHERE ? </query>
        <query tableName="store">SELECT * FROM store WHERE ? </query>
        <query tableName="store_ragged">SELECT * FROM store_ragged WHERE ? </query>
        <query tableName="time_by_day">SELECT * FROM time_by_day WHERE ? </query>
        <query tableName="warehouse">SELECT * FROM warehouse WHERE ? </query>
        <query tableName="warehouse_class">SELECT * FROM warehouse_class WHERE ? </query>
      </queries>
    <periodicity>5000000</periodicity>
  </task>
</tasks>
```

Figure 2. An excerpt of a configuration file of a data provider

To manage and control the system, we created two other agents: an administrator and a task monitor. Both are system administration agents. The first one is responsible to activate or terminate the multi agent system, keeping it active, managing the insertion or removal of other agents in the system (Fig. 3).

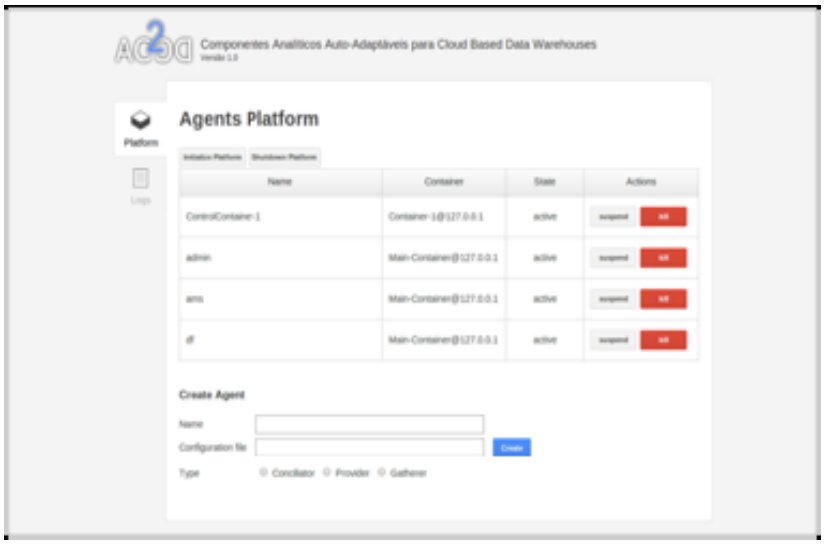

Figure 3. A list with all active agents in the system

The monitor has the task to report all activities that system's agents are doing. All system's software agents were developed in Java and their community environment for data sharing communication and collaboration was supported by the framework JADE [27] following the standard specifications of FIPA (www.fipa.org).

3.3 The Personalization Process

In order to demonstrate how agents automatically adapt the data visualization structures, we developed a Web based application to receive the main system interface platform. On this platform we integrated six data visualization components in a single dashboard, having six different segments of analysis, namely: two data grids, three charts and one geo chart (Fig. 4). All data visualization components were developed using the Google Chart Tools [28]. These tools provide a large diversity of interactive and personalized graphical elements that can be used on Web browsers quite autonomously without having the need of any other computational means. The initial state of the dashboard is defined by a predefined configuration that was communicated to the Web application by a data visualizer agent that assumed the control of the analytical session. The dashboard was prepared according to the analytics preferences defined previously by the user, using proper credentials. The way a user acts on an interface platform depends on the functionalities of the graphical elements located in the several segment of analysis.

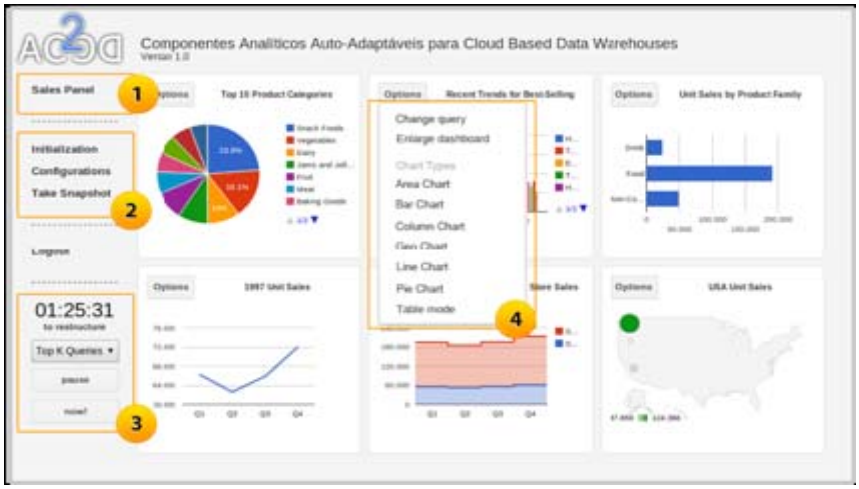

Figure 4. A system's interface platform

In Fig. 4 we can see several labels – 1, 2, 3 and 4 – that indicate the options we have to configure an interface platform and the graphical elements located in. These options allow for users to personalize each one of the segments of analysis. Label 1 shows all the dashboards configurations that a user has to his disposal. Selecting one of the available configurations, automatically the dashboards changes showing the correspondent segments of analysis of the configuration selected. Labels 2 and 3 show the places where we can manage globally the dashboards that were defined. In the options of Label 2, we can load the initial configuration of a dashboard (Initialization), to list all the configurations defined for a specific dashboard (configurations) (Fig. 5), and take a snapshot of the current dashboard storing it to be used later if necessary.

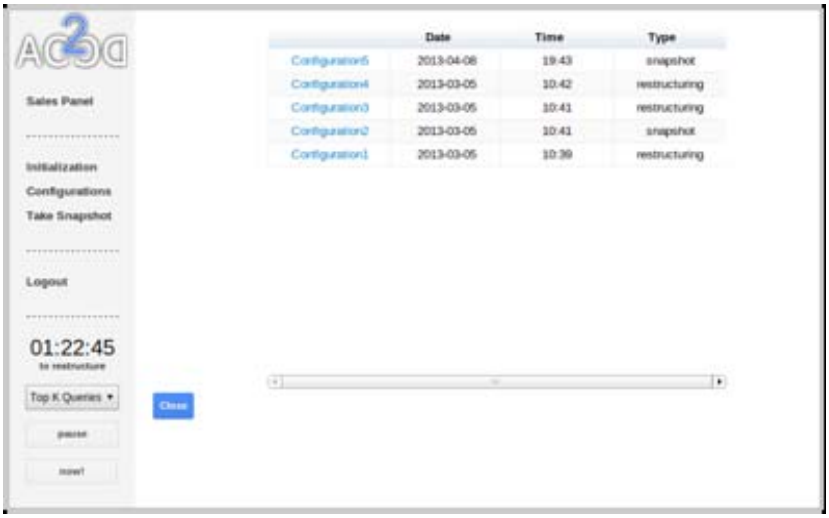

Figure 5. A configuration list for a dashboard

Label 3 indicates the place where we can select the method for restructuring (e.g., top-k queries, association rules, or Markov chains) a specific dashboard in order to personalize it. The restructuring period of a dashboard is set previously in the agenda of each data visualization restructurer. Finally, label 4 reveals the configuration options of the individual graphical elements – e.g., changing the query, enlarging the dashboard, or defining the chart type. During an analytical session, users can explore all the analysis segments that are available in their current dashboards, using their own functionalities (changing the data visualization mode) or submit new queries through a specific querying interface (changing or refreshing data). All these operations are stored on specific event logs.

Dashboards restructurers, data visualizers and data providers support the entire personalization service. The analysis of the log's contents is done exclusively by the dashboards restructurers, which suggest (when possible) a new configuration for the dashboard. Then, the new configuration is communicated to a data visualizer that informs posteriorly the user about it – the user may accept or refuse this new dashboard configuration. As already said, dashboards restructurers use several strategies to evaluate and suggest new configurations for dashboards.

4. Conclusions And Future Work

The personalization of an analytical system is not a simple task. Acquiring usage preferences or extract some valid knowledge from an analytical session are tasks that are difficult to perform. However, it is possible to change some of the most conventional features of an analytical system, in order to make it look more "intelligent" doing things in an expert way. Using a specialized community of agents, we developed an agentbased tool that has the ability to personalize analytical dashboards based on the preferences and tendencies that users showed during their past analytical sessions. User preferences, regular analytical operations, and querying sequences were the "raw material" we need to personalize analytical systems and to allow for creating self-adaptation mechanisms for analytical dashboards. The use of agents revealed to be an adequate choice. They supported well concurrent operations and received appropriately new agents every time it was necessary to setup a new interface platform. However, for now, this is only a prototype system. We need to test and evaluate the system in a real-world scenario using a large number of users and greater volumes of data. Additionally, we need to design and implement new and better algorithms for user

preferences acquisition and analytical usage profiling.

5. Acknowledgment

This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

[1] Slaughter, A. (2015). OLAP Introduction, 2004. [Online] Available at: http://web.mit.edu/profit/PDFS/SlaughterA .pdf [Accessed 13 December 2015].

[2] Few, S. (2006). Information Dashboard Design. Sebastapol, California: O'Reilly Media, Inc.

[3] Billsus, D., Pazzani, M.J. (2000). User Modeling for Adaptive News Access. Irvine, CA, USA: Kluwer Academic Publishers Dept. of Information and Computer Science, University of California.

[4] Mezhoudi, N. (2013). User Interface Adaptation based on User Feedback and Machine Learning. Louvain: Université Catholique de Louvain.

[5] Bellatreche, L., Giacometti, A., Marcel, P., Mouloudi, H., Laurent, D. (2005). A Personalization Framework for OLAP Queries, *In*: Proceedings of DOLAP'05, Bremen, Germany, November 4–5.

[6] Kozmina, N., Niedrite, L. (2010). OLAP Personalization with User-Describing Profiles. In Forbrig, P., Günther, H. (eds.) BIR 2010. LNBIP, vol. 64, p. 188–202. Springer, Heidelberg

[7] Wooldridge, M. (2002). An Introduction Multi-Agent Systems. Jonh Wiley & Sons, Ltd

[8] Wooldridge, M., Jennings, N. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10 (2) 115–152.

[9] van der Hoek, W., Wooldridge, M. (2008). Multi-Agent Systems, In F. van Harmelen, V. Lifschitz, and B. Porter, editors, Handbook of Knowledge Representation, p. 887—928. Elsevier.

[10] Müller, J., Fischer, K. (2014). Application Impact of Multiagent Systems and Technologies: A Survey. *In*: Onn Shehory, Arnon Sturm (eds.) Agent-Oriented Software Engineering, p. 27-53, Springer-Verlag.

[11] Blewett, R. (2011). The Importance of Data Visualization to Business Decision Making, Dundas Data Visualization, Inc., June 12.

[12] Spence, I., Playfair, William (1759–1823). Inventor of statistical graphs and writer on political economy, 2004. [Online] Available at: http://www.psych.utoronto.ca/users/spence/Spence_2004.pdf [Accessed 13 December 2015].

[13] Wallis, J. Databases & Data Visualization: the State of the Art, CENS, UCLA, 2005. [Online] Available at: http://www.moebiustrip.org/CENS/DataVisTechRep.doc [Accessed 31 May 2015].

[14] Friendly, M. (2006). A Brief History of Visualization. *In*: Handbook of Computational Statistics: Data Visualization, Vol. III, Heidelberg , Springer-Verlag

[15] Corbett, J. Charles Joseph Minard: Mapping Napoleon's March, 1861. [Online] Available at: http://www.csiss.org/classics/content/58 [Accessed 13 December 2015].

[16] Few, S. Data Visualization Past, Present and Future, 2007. [Online] Available at: http://www.perceptualedge.com/articles/Whitepapers/Data_Visualization.pdf [Accessed 4 August 2016].

[17] Tufte, E. (1983). The Visual Display of Quantitative Information. 2nd ed. Cheshire: Graphics Press.

[18] Anke, J., Sundaram, D. (2006). Personalization Techniques and their Application, *In*: Khosrow-Pour, M., Ed., Encyclopedia of E-Commerce, E-government and Mobile Commerce, Hershey, PA: Idea Group Reference

[19] Freedman, L. (2007). Merchant Views of personalization and lasting customer relationships. Special report, ATO, October

[20] Eirinaki, M. Vazirgiannis, (2003). Web mining for web personalization, *ACM Trans Inter Tech* 3. p. 1-27.

[21] Harris, L., Baney, D., Davis, B., Nelson, H. (2005). The Personalization of Data for Print and e-Commerce, Arizona State

University.

[22] Bhatia, D., Nirkhi, P. Bajaj, S. (2011). Context-aware Personalized Mobile Web Search Techniques-A Review, *International Journal of Computer Science and Information Technologies*, 2 (5) 2440-2443.

[23] Petit-Rozé, C., Strugeon, E. (2006). MAPIS, a multi-agent system for information personalization, *Information and Software Technology* 48 (2) 107–120, Elsevier, February.

[24] Thomas, C., Fischer, G. (1997). Using agents to personalize the Web. *In*: Proceedings of the 2nd international conference on Intelligent user interfaces (IUI '97), Johanna Moore, Ernest Edmonds, and Angel Puerta (Eds.). ACM, New York, NY, USA, 53-60

[25] Kazienko, P., Kiewra, M. (2003). ROSA – Multi-agent System for Web Services Personalization. *In*: E. Menasalvas et al. (Eds.): AWIC 2003, LNAI 2663, p. 297–306, 2003.Springer-Verlag Berlin Heidelberg.

[26] Wood, M., DeLoach, S. (2001). An Overview of the Multiagent Systems Engineering Methodology, *In*: Agent-Oriented Software Engineering – Proceedings of the First International Workshop on Agent-Oriented Software Engineering, 10th June 2000, Limerick, Ireland. P. Ciancarini, M. Wooldridge, (Eds.) Lecture Notes in Computer Science. Vol. 1957, Springer Verlag, Berlin, January 2001.

[27] Bellifemine, F., Caire, G., Greenwood, D. (2007). Developing Multi-Agent Systems with JADE, John Wiley & Sons Ltd.

[28] The Google Chart Tools. [Online] Available at: https://developers.google.com/chart/ [Accessed 4 August 2016].