An Approach for a Web-based Analysis Solution with MUSTANG

Mirco Josefiok* and David Korfkamp† and Jan Witt‡

OFFIS – Institute for Information Technology
Data Management and Data Analysis
Escherweg 2, 26121 Oldenburg
Email: †josefiok@offis.de, †korfkamp@offis.de, ‡witt@offis.de

***Abstract*—Analytical Information Systems (AIS) are comprehensive solutions for analyzing large data sets. The operation of an AIS usually requires an extensive infrastructure. Moreover, usually only specialist users are capable of performing analyses. In this paper, we present an approach for a web-based analysis solution which can be deployed either in a web-based environment or as an on-premise solution. We strongly emphasize self service capabilities by adding knowledge-based components in the form of an additional metadata layer.**

Keywords–web-based system; analysis information systems; self-service business intelligence.

I. Introduction

For coping with an increasing flow of information in companies and a growing complexity in planning business decisions, Data Warehouses (DWH) with Online Analytical Processing (OLAP) have been established. With these technologies and concepts, a company-wide provision of information should be guaranteed. Analytical Information Systems (AIS) are the logical bracket around the concepts and technologies DWH, OLAP, Data Mining and the respective analysis tools [1].

Development and operation of an infrastructure for processing, storing and analyzing large amounts of data requires substantial investments. To be able to react promptly and appropriately and to acquire new data whenever available, the necessary capacities must be maintained. Therefore, small and medium enterprises, as well as other entities whose core competencies are not information technology, do not use dedicated analysis platforms or business intelligence solutions commonly [2].

Analytical Information Systems (AIS) (see section II-B) usually suffer from some shortcomings when it comes to self-service of business users. Notably, typical business users are not capable of performing analyses by themselves, because the software solutions are too complex and lack the necessary guidance. Moreover, the underlying data model is often too complex to comprehend. Usually, no metadata is present, which might help business users to gain insights about structure and semantics of the underlying multidimensional data [3].

This raises the demand for a solution which is easily accessible, scales with an increasing amount of users but does not lack the analytical capabilities specialist users demand.

In this paper, we present a web-based approach for an analysis solution. Our approach is based on the MUSTANG (Multidimensional Statistical Data Analysis Engine) framework [4] and is developed within the WAIS project [5]. MUSTANG is used, for example, by epidemiological cancer registries (ECRs) in several German federal states in an instance called CARESS [6]. Whilst MUSTANG and its instances were originally developed as standalone applications, recent changes made it possible to deploy it as a service-based application in cloud environments. This results in the opportunity to make the analysis solution available to a broader audience, which is backed up by the new approach of the presented solution. With curated data, even more value can be added to our approach. Moreover, we introduced an additional metadata layer to foster self-service operations.

This paper is organized as follows. Section II introduces the foundations of our work, namely Software as a Service (SaaS), AIS and the MUSTANG framework. Section III analyzes the problem statement and introduces the architectural requirements for a service-based AIS. Section IV gives an overview of selected related works. Section V presents our approach and gives an overview of the developed prototype and its architecture. Section VI describes the evaluation and points out remaining research and development challenges. Finally, the paper concludes with Section VII.

II. Foundations

Cloud computing is currently considered one of the most important topics for the information and communications technology (ICT) sector which has emerged in recent years. It already has a significant impact on how most IT-related projects are pursued, especially regarding the design and implementation of new software products. In addition, increasing innovation may be possible with the new deployment options available through cloud computing [7].

Several conceptual frameworks to describe and characterize cloud service offers exist. As they are developed by different groups and organizations, they differ in their intention, type of formulation, the level of description, and in terms of which key issues of cloud service evaluation are addressed. The American National Institute of Standards and Technology (NIST) provides a definition of cloud computing which is well accepted in industry and science [8]. A recent survey of different frameworks and reference models for the description of cloud offerings and taxonomy of cloud service offers is provided by Gudenkauf et al., with the aim to help overcome the skepticism of enterprises regarding cloud service offers by identifying their key factors [9].

A. Software as a Service (SaaS)

According to Babar and Chauhan [10] SaaS can be defined as follows: "The capability provided to the consumer [. . .] to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings." Our approach can be regarded as a special form of SaaS, in which analytical systems are deployed at a hosted location and accessed by potential users with regular internet connections.

For a potential user, this model has a wide range of benefits. Besides the benefits which result from the usage of cloud services, like cost alignment, cost reduction, streaming payment and compliance implementation, potential end users can focus on their core competencies and do not need to bother about building and maintaining a rather complex infrastructure for running an AIS.

B. Analytical Information Systems (AIS)

AIS have been established in science and industry for performing analyses on large data sets. AIS should support analytical activities by summarizing different concepts and technologies (DWH, OLAP, Data Mining, Business Intelligence solutions) and present them to the user due to a unified view. The DWH acts as central database in which all relevant data should be integrated via ETL processes (extract, transform, and load). With OLAP a multi-perspective view on the data can be enabled. Data Mining is an umbrella term for different methods in the field of data analysis. End users have access to those concepts and methods via business intelligence tools. AIS provide those concepts, methods and tools to end users and support them in the process of information retrieval and decision making [1].

C. Self-Service Business Intelligence

Self-service business intelligence (SSBI) can be defined as a BI environment, which empowers business users to perform complex analyses without needing the help of an expert [11]. This allows users to acquire information in a more timely manner and makes them more independent from dedicated BI departments. Major requirements for establishing a SSBI solutions are understandable and comprehensible results, accessible software solutions, fast response times when performing complex analyses and quick and easy access to the source data.

D. MUSTANG

MUSTANG is a framework for creating AIS. The system supports non-technical users by providing tools for data analysis in a highly accessible user interface, enabling them to carry out explorative analyses, ad-hoc queries and reporting activities. A usual MUSTANG application consists of three layers. The Data Integration Layer provides a unified physical integration of various heterogeneous data sources, e.g., data warehouses, metadata and geographic information systems. The Component Integration Layer provides services

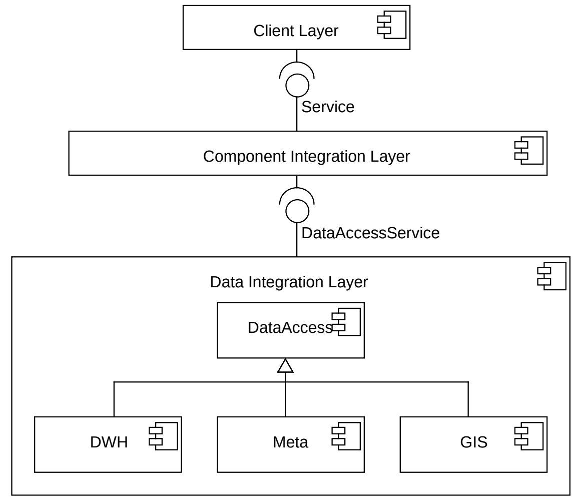

Figure 1. MUSTANG Architecture

to perform different types of analyses and to expose metadata structures like saved analysis configurations and the structure of the underlying data warehouse. The client layer provides an user interface (UI) which orchestrates the services of the Component Integration Layer and visualizes their results. By now only a desktop rich client existed - in the course of this work we developed a new web-based client described in the upcoming sections. Figure 1 shows a simplified overview of the MUSTANG architecture.

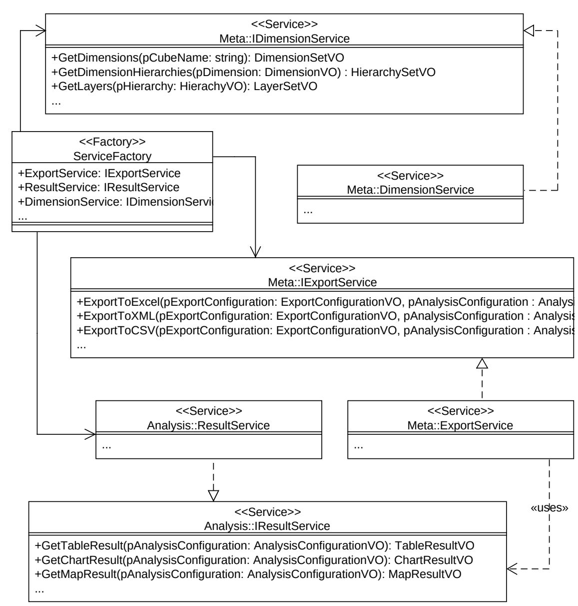

Figure 2. Services classes of the component integration layer

The individual layers are encapsulated from each other. Especially the component integration layer provides a service facade to potential client applications. Figure 2 shows an example for the services classes of the component integration layer. The figure depicts three exemplary MUSTANG services

 ISBN: 978-1-61208-387-2

as well as the service factory which maintains instances of all MUSTANG services. As visible in the figure, all services in MUSTANG are written against interfaces and the service factory exposes only these interfaces in order to maintain interchangeability of services. The central artifact of a MUSTANG analysis is the *AnalysisConfigurationVO* which is used across all services of the Analysis namespace. The exemplary services *ExportService* and *ResultService* use this object to calculate analysis results and provide it for visualization or to export it into various formats. The *DimensionService* is used to expose the dimension structure of the underlying data warehouse which is one piece of an analysis configuration in MUSTANG [6].

III. Key Business Drivers and Requirements

Migrating an existing software system to a cloud environment or even providing it as a service is a very difficult task. First, it has to comply with the service-oriented architecture (SOA) paradigm. A major advantage of doing this is, that with a SOA multiple services can be orchestrated for higher-value services [12][13]. Cloud based applications can be considered a collection of services. Software systems which have been implemented in a service oriented way, should be able to adopt cloud computing more easily [14].

AIS are usually very complex systems, dealing with various different data sources and varying user groups. In most cases, only specialist users conduct analyses with AIS. With this in mind, in the course of the project, the following business drivers and requirements could be identified. Several one-day workshops with potential users were held. The following list gives an overview of the most relevant business drivers and requirements.

Multidimensional analyses: Multidimensional analyses capabilities are required to allow the exploratory analysis of complex data.

Performance: Performance comparable to the CARESS Desktop Client [6] is desired.

Modularity and expandability: It should be possible to subsequently add complementary functionality to the prototype.

Web standards: The prototype should use web standards and should conform to the RESTful architecture paradigm [15].

Metadata: Meaningful metadata should be provided to increase efficiency and accessibility when performing analyses.

User empowerment: It should be possible for inexperienced users to perform complex analyses.

We will address the implementation of each individual business driver in Section V.

IV. Selected Related Work

In the field there exist several approaches that address the aforementioned requirements more or less. In this section we present a selection of related workings and measure their degree of fulfilment of these requirements. This selection consists of CARESS, an instance of MUSTANG with a strong focus on the requirements of epidemiological cancer registries in Germany [6], Tableau, a software tool for analyzing different data sources of all kinds, focusing on self-service BI [16], KNOBI (Knowledge-based Business Intelligence), which is an instance of MUSTANG enriched with an ontology-based semantic layer knowledge [11] and Super Data Hub a web-based analysis solution for big data [17]. While the requirement *multidimensionality* has been met by all related approaches, the other requirements are only fulfilled partially which is especially true for *Metadata* and *Web standards*. See table I for an overview of the related workings and their requirements coverage.

	CARESS	**Tableau**	**KNOBI**	**Super Data Hub**
Multidimensional analyses	x	x	x	x
Performance		x		
Modularity and expandability	x		x	
Web standards				x
Metadata	x		x	
User empowerment		x	x	x

TABLE I. Requirements coverage of related work

V. Approach

In this section, we present our approach for a web-based analysis solution based on the MUSTANG framework. In Section V-A, we describe the server side of our application and especially how we utilize the MUSTANG framework for our purpose. In section V-B, we explain the implementation of our client with particular emphasis on self-service and the usage of metadata.

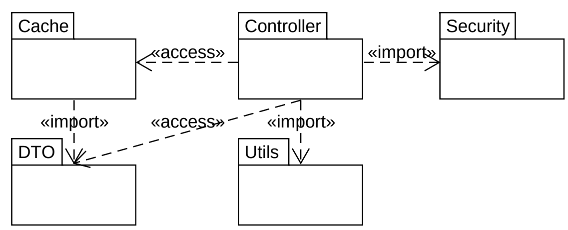

Figure 3. Server Packages

MUSTANG itself started with a monolithic design, but over the course of the past two years it was reengineered towards a more service-oriented architecture. The system itself was divided into loosely coupled components which access each other through a newly added service layer. Moreover, this RESTful service layer was added to encapsulate view logic even further. This allows using different existing cloud services for hosting different parts of the system. Figure 1 shows a simplified architecture of a possible system based on MUSTANG. Each data source, the core application and the view or client application may be deployed using different cloud services.

A. Prototypical Application - Server Side

On foundation of the reengineered MUSTANG framework we developed a prototypical application. The MUSTANG framework already offers rich functions for multidimensional data analysis that we utilized in order to contribute to the business driver *Multidimensional analyses*. We added a completely

 ISBN: 978-1-61208-387-2

new developed HTML5/JavaScript frontend to the desktop application. Therefore, we relied on RESTful web services for data exchange between client and server. We decided to use these technologies in order to comply with the business driver *Web standards*. Moreover, we added the capability to work securely in a multi-client environment.

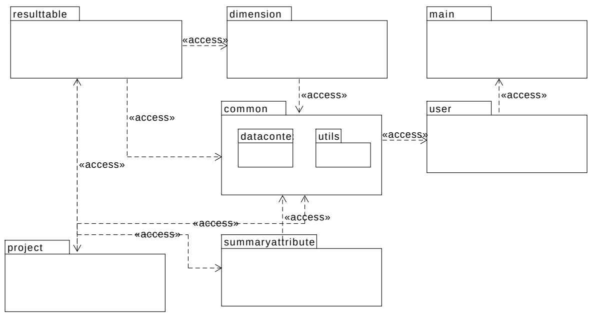

Figure 4. Client Structure

Figure 3 shows the package structure of the server-side application. It is developed on top of the component integration layer of the MUSTANG framework and provides an interface for accessing multidimensional data analysis via web services. For implementing the application, we used the Microsoft ASP.NET Web API 2 which offers an easy way to create RESTful web services. Our application is structured into different packages. The main point is the *controller package*, which contains the controller-classes. A controller class consists of methods which handle and answer the incoming client's HTTP requests (GET, POST, DELETE, PUT). The returning values are mostly in the form of lightweight data transfer objects (DTO) which are organized in the *DTO package*. A DTO in our application corresponds to a *Value Object* provided by the MUSTANG framework. We tailored our DTOs to have a smaller size compared to the corresponding Value Object by removing all information not required by our application. This way we gained quicker response times from communication between web browser and ASP.NET server contributing to the business driver *Performance*. Before transferring the DTOs to requesting clients, they are serialized to the JavaScript Object Notation (JSON), which is the preferred data exchange format when implementing REST APIs.

Another contribution to the business driver *Performance* is the reduction of response times for sending requested data by our implemented caching functions for OLAP-Server-access, which are concentrated in the *Cache package*. Especially the time for retrieving measures, classification nodes and comprehensive metadata-information could be reduced from minutes to seconds by instantiating them already on server start through our caching mechanism, which makes the objects present in-memory. The *Security package* holds all logic necessary for authentication and user and group management. At last, the *Utils-package* contains various utility-classes for handling and transforming multidimensional data.

B. Prototypical Application - Client Side

Our client application consumes the web services of the described server side application. It was developed using HTML5/JavaScript, especially AngularJS [18]. In order to contribute to the business driver *Modularity and expandability*, the application is organized in a modular way with reusable components representing different aspects of the view. Figure 4 gives an overview of the client application's structure.

The *main module* represents the applications starting point. It accesses the *user module* and therefore log-in, log-out and client side security logic. The central module is the *common module* which is accessed by all other modules because its *datacontext* holds the overall data which is re-used throughout the application, e.g., selected multidimensional data, analysis metadata and user data.

The *measures module* contains everything needed to select measures. Assisted by comprehensive metadata, the users can scan a measure-list by entering search terms or using filtering mechanism based on facet classifications to identify proper measures for their analysis intention. The metadata for measures include, for example, information about theme, origin of data, area coverage or available dimensions. In this way, the business drivers *Metadata* and *User empowerment* are addressed.

Figure 5 shows the modal window for selecting measures. 1 shows the facet classification. Facets can be filtered by name and description and individually grouped. The facet definition is part of the Extract-Transform-Load (ETL) process. Users are able to browse, select and deselect different facets. 2 shows list of available measures. For each measure, the most important meta data is shown right away while additional information can be accessed via a tool tip. 3 shows the selected measures as well as the dimensions available with these measures.

In the *dimension module* the functionality for selecting dimensions and appropriate classification nodes is located. Here, the user can navigate by choosing fitting layers or by expanding a hierarchy to find suitable classification nodes. A function to merge different nodes for an ad-hoc aggregation is included as well. Again, it is possible to display additional metadata for individual elements.

The main point for analysis purposes is organized in the *resulttable module*. After selecting measures, dimensions and classification nodes the resulting table is generated here. The user can manipulate the table via drag and drop in an interactive way. Dimensions and measures can be moved and arranged in columns and rows. For a better table view, it is possible to change to a full screen mode, in which all nonessential information is faded out. Furthermore, functionality to export table data into Excel or comma separated value (CSV) format is included. Figure 6 shows a generated table. 1 shows user interface elements for manipulating the table (e.g. moving dimensions from one axis to another). 2 shows the table itself and the result data. Finally 3 shows the selected measures and the user interface elements to determine the presentation of the measures.

The functionality to save analysis data is implemented in the *project module*. Analyses can be organized in collections called projects and enriched with metadata like descriptions, editors and visibility declarations (private vs. public).

 ISBN: 978-1-61208-387-2

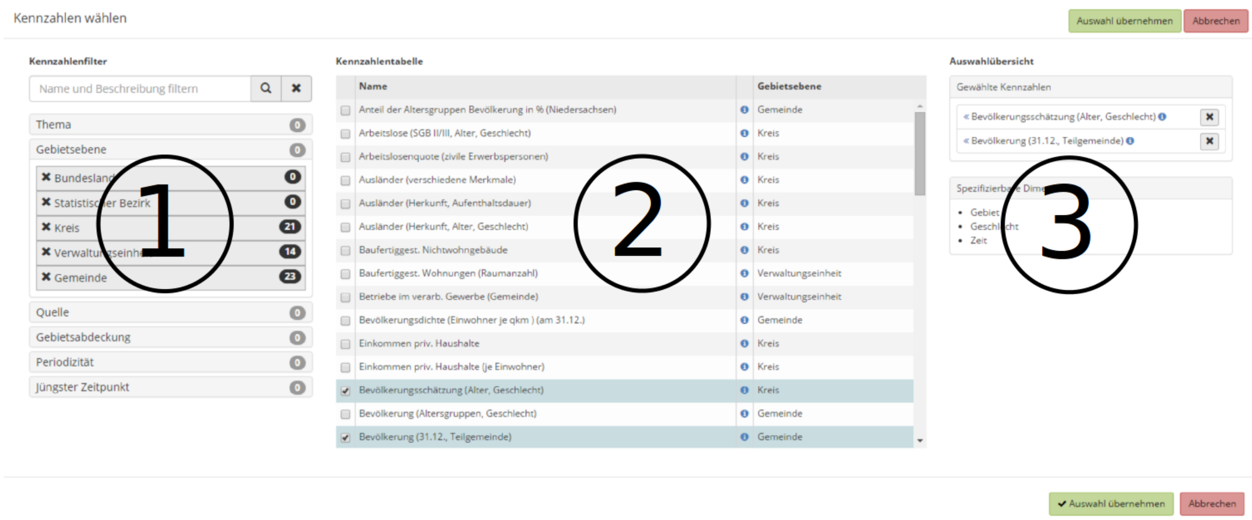

Figure 5. Measure Selection

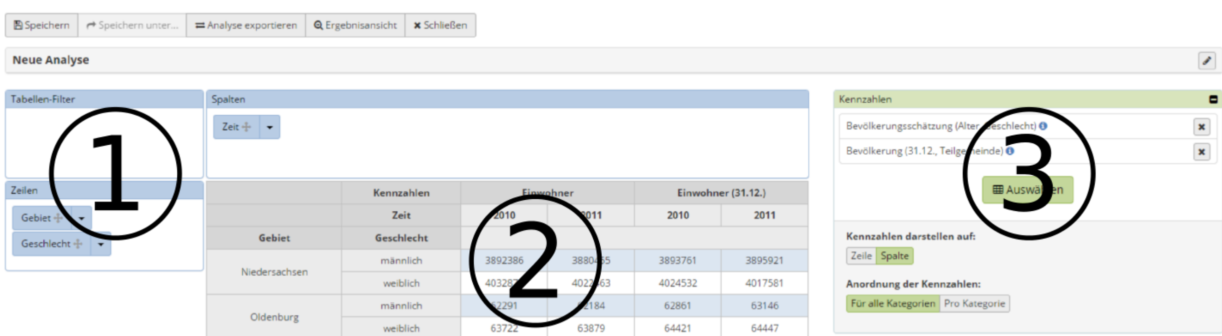

Figure 6. Result Table

VI. EVALUATION AND REMAINING CHALLENGES

The application has been shown to potential users on various occasions, but an extensive evaluation is currently underway and will be completed by the end of 2014. By now, the application was evaluated on four occasions with individual users or user groups. A scenario was created for the evaluation which included a typical task with a corresponding analysis questioning. The participants were encouraged to solve the task and answer the question with the system at hand. Each evaluation lasted about two hours with concluding interviews and a structured feedback process. The evaluation was recorded and will be processed and evaluated step by step. Right now only a first impression from the first evaluation appointments can be given.

The first group consisted of specialist users who are experienced in working with web-based analysis solutions and have an extensive knowledge of the analysis domain. The second group consisted solely of business users with an extensive knowledge of the domain but minor experience in working with analysis solutions. Whilst the first group does not represent the desired core audience for the application, their feedback is also valuable for improving the analysis capacity.

In preparation of the evaluation, an internal evaluation was performed. A colleague who was not associated with the project performed the evaluation. The purpose of this proceeding was to evaluate the scenario, the task to solve and to find bugs or stumbling blocks in the application itself.

Feedback from the first group has been mostly positive. As the majority of participants knew about the domain and the structure of the multidimensional data, they were able to produce valid results very quickly and to solve the given tasks. All agreed that the provided metadata allows performing valid analyses in a convenient way and producing valuable results faster. Some struggled with the concepts of facets in the dialog for selecting measures. This concept — to our knowledge — is not implemented in other applications so far. Metadata can be used in this case for filtering a measure-list. Some participants thought they could already select and specify dimensions while selecting measures. Not all provided metadata was directly accessible and was not used in the first place.

Feedback from the second group has been mostly positive as well. Before evaluation started, a short introduction was given to each participant. Some participants struggled with the overall operation of the system but were able to solve the given task with minimal help. Some participants observed inconsistencies in the operating concept regarding the specification of

 ISBN: 978-1-61208-387-2

measures.

An initial conclusion which we can derive from the evaluation so far is that our overall concept works well and the provided metadata is helpful for possible users with or without experience in working with analysis solutions. But we also found some misconceptions and inconsistencies in our operation concept, especially regarding the provisioning of even more metadata and the concept of facet classification.

VII. Conclusion and Future Work

We presented an approach for a service-based analysis solution with self-service in mind. Our approach proves to be an improvement over existing solutions. There are still a number of issues to be addressed before the solution can be used on a daily basis. We showed that our solution is accessible by business users with only minimal instructions while we preserve the analysis capabilities of an analytical information system.

Up to now we deployed the application only in a local environment. In addition, we only performed limited load tests. Therefore, we need to make sure our application scales with an increasing amount of users simultaneously performing analyses. Whilst MUSTANG is a proven and robust framework it was not tested yet in a multi-user environment.

In Section I we mentioned two main shortcomings of common analysis solutions. We addressed the issue, that typical business users are not capable of performing analyses by themselves by creating a highly accessible solution. Our solution heavily utilizes supporting metadata, which addresses the problem, that the underlying data model is too complex to comprehend. In further iterations, we plan to extend the usage of metadata and knowledge-based approaches. Our system is therefore designed and implemented in an extensible way.

This should enable business users to perform analyses without extensive training or the aid of specialist users. The evaluation showed, that in many cases metadata is sufficient, but in some cases business users require more guidance. This applies especially if they only perform analyses at fixed times (e.g., once per month or year) but not regularly. With a little more work we are sure our approach can be understood as a SSBI solution.

We mentioned some issues in Section VI, mostly regarding our operation concept. Moreover, we did not yet take security issues into account. Whilst we developed only a research prototype, a possible application which will be accessible by end users must take security issues into account. In addition, we will rework parts of our operation concept, especially regarding the facet classification as part of the measure selection.

Another planned improvement concerns the contribution of user-specific data. At the moment, it is only possible to perform analyses with the data available in the underlying DWH. A future version could incorporate an interface with an associated process which allows user groups to add their own data to a user-specific location. Particularly, this is important because certain data is not freely available and must not be available to users outside a respective group.

Acknowledgment

This work is supported by the Federal Ministry of Education and Research (BMBF) on the basis of a decision by the German Bundestag under Grant No. 01IS12042B.

References

[1] P. Chamoni and P. Gluchowski, “Analytische Informationssysteme - Einordnung und Überblick,” in Analytische Informationssysteme : Business Intelligence-Technologien und -Anwendungen, 2010, pp. 3–16.

[2] M. Mircea, B. Ghilic-Micu, and M. Stoica, “Combining business intelligence with cloud computing to delivery agility in actual economy,” Journal of Economic Computation and Economic Cybernetics Studies, vol. 45, no. 1, 2011, pp. 39–54.

[3] M. Mertens, T. Krahn, and H.-J. Appelrath, “Utilizing Structured Information from Multiple External Sources in the Context of the Multidimensional Data Model,” in BIS 2013, ser. Lecture Notes in Business Information Processing,, Abramowicz, Witold. Heidelberg: Springer, 2013, pp. 88–99.

[4] O. I. for Information Technology, “Multidimensional statistical data analysis engine,” http://www.offis.de/en/r_d_divisions/health/project/projekte/mustang.html, last visited 2015-02-09.

[5] ——, “Wais - aufbau eines wissensbasierten analytischen informationssystems für die kollaborative datenanalyse,” http://www.offis.de/en/r_d_divisions/health/project/projekte/wais.html, last visited 2015-02-09.

[6] D. Korfkamp, S. Gudenkauf, M. Rohde, E. Sirri, J. Kieschke, and H.-J. Appelrath, “Opening up Data Analysis for Medical Health Services: Cancer Survival Analysis with CARESS,” in Data Warehousing and Knowledge Discovery - 16th International Conference, Munich, Germany, 2014.

[7] P. Hoberg, J. Wollersheim, and H. Krcmar, “The Business Perspective on Cloud Computing-A Literature Review of Research on Cloud Computing,” AMCIS 2012 Proceedings, no. 5, 2012.

[8] G. Vossen, T. Haselmann, and T. Hoeren, Cloud-Computing für Unternehmen: Technische, wirtschaftliche, rechtliche und organisatorische Aspekte. Heidelberg: dpunkt.verlag GmbH, 2012.

[9] S. Gudenkauf, M. Josefiok, A. Göring, and O. Norkus, “A Reference Architecture for Cloud Service Offers,” in EDOC 2013, 2013.

[10] P. Mell and T. Grance, “The NIST Definition of Cloud Computing - Recommendations of the National Institute of Standards and Technology,” National Institute of Standards and Technology, Gaithersburg, Tech. Rep., 2011.

[11] M. Mertens, KNOBI - Knowledge-based Business Intelligence for Business User Information-Self-Service -, 1st ed. Oldenburg: OlWIR Verlag für Wirtschaft, Informatik und Recht, 2013.

[12] G. Feuerlicht, L. Burkon, and M. Sebesta, “Cloud Computing Adoption: What are the Issues,” Systems Integration, 2011, pp. 187–192.

[13] M. A. Babar and M. A. Chauhan, “A tale of migration to cloud computing for sharing experiences and observations,” in Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing, ser. SECLOUD ’11. New York, NY, USA: ACM, 2011, pp. 50–56. [Online]. Available: http://doi.acm.org/10.1145/1985500.1985509

[14] J. Lawler, “The Potential Reality of Service-Oriented Architecture (SOA) in a Cloud Computing Strategy,” Journal of Information Systems Applied Research, vol. 4, no. 1, 2011, p. 57.

[15] S. Kumaran, R. Liu, P. Dhoolia, T. Heath, P. Nandi, and F. Pinel, “A restful architecture for service-oriented business process execution,” in Proceedings of IEEE International Conference on e-Business Engineering, 2008, pp. 197–204.

[16] T. Software, “Fast analytics and rapid-fire business intelligence from tableau software,” http://www.tableau.com/, last visited 2015-02-13.

[17] SuperDataHub, “Superdatahub - cloud bi - superdatahub,” http://superdatahub.com/, last visited 2015-02-13.

[18] Google, “Angularjs — superheroic javascript mvw framework,” https://angularjs.org, last visited 2014-10-28.

 ISBN: 978-1-61208-387-2