УДК 665.9
DOI: 10.15587/2313-8416.2015.36747

THE OVERALL STRUCTURE OF THE WEB DATA MINING REPOSITORY

© T. Shatovska

The article discusses an approach of multi-agent system building which allow users to work with statistical data. Repository is based on the idea of ontological models implementation and idea of their interaction through the set of the intelligent agents. This approach will allow a meaningful way to select a required datasets according to the given problem domain, as well as add new data sets using their meta descriptions

Keywords: *Repository, Data Mining, Ontology, Multiagent system, Intelligent agent, Database, Jadex*

У статті розглядається підхід многоагентних побудови систем, який дозволяє користувачам працювати зі статистичними даними. Репозиторій заснований на ідеї онтологічної реалізації моделі та ідеї їх взаємодії через набір інтелектуальних агентів. Такий підхід дозволить осмислено вибрати необхідні набори даних по заданій предметній області, а також додавати нові набори даних, використовуючи їх метаописания

Ключові слова: *репозиторій, Data Mining, онтологія, багатоагентна система, інтелектуальний агент, база даних, Jadex*

1. Introduction

We envisage a world where the barriers to sharing and exchanging data and information are radically increased. The world we are undoubtedly moving toward is one of Web-based ‘mash-ups’; that is, networked software applications that can combine data in real-time from multiple service providers in ways that are user-friendly, yet powerful. To be effective in this space, it is imperative that repositories become first-class service providers.

Repository is a dataset of statistical data, which contains data, data description, and metadata description to them. This involves collecting and preserving good metadata. Hence, metadata is not simply about technical requirements specific to repositories; rather, it forms the basis of an emerging information infrastructure for data stores communications that has far-reaching consequences.

Digital repositories are networked software applications primarily used for storing, managing and disseminating data (e.g. digital publications, theses, data sets and so on). The Repositories differ from conventional content management systems because they include technologies to ensure that data are preserved for long-term access and use. Although repositories were initially developed for science purpose, statistical companies, they are currently being implemented more widely; for example, by museums to facilitate online access to cultural heritage resources, and government agencies to mediate long-term access to documents and other data. In practical terms, implementing a digital repository nowadays can be as simple as downloading free open-source software and installing it onto a networked computer.

2. Literature review

Establishing a stable repository for everyday institutional use is an altogether harder proposition. The most popular open source repository applications are UCI Knowledge Discovery in Databases Archive [1, 2]; DEA Dataset Repository, Frequent Itemset Mining Dataset Repository, XMLData Repository [3–6], etc. There are some commercial repository software providers, but none have gained the same level of popularity as the open source repositories mentioned above. The important point to note here is that a repository is essentially a relational database that stores and keeps track of metadata records for files stored ina mass-data storage facility.

The underlying technology is relatively straightforward whereas the institutional context of use is typically complex. These systems are not information. It is difficult to exchange files automatically.

The main reason to create a new public information repository is to improve structure of repositories where datasets stored using intelligent agents and ontology approach for storing, conversion, search, add, description, selection of the required information for researchers needs. We focused on the development of the ontology models for data mining methods, ontology model for data transformation methods and intellectual search agent for collaboration between these models, datasets and user profile in the field of Data mining and Machine Learning datasets. In this paper presents the overall structure of the web data mining repository.

3. W-DMR

Let us consider some stages of the system implementation.

First of all repositories of scientific collections of statistical data, which identified their strengths and weaknesses were deeply researched and analyzed. This analysis has helped on the basis of the analysis characteristics of the shortcomings of existing repositories to develop a software implementation. It solves the problem of preservation of large and stable datasets using ontological models in the hierarchical structures and improves the efficiency of working with them. The Ontology is a complete structural specification of a certain subject area, its formal submission, which includes a glossary of terms of that area and the set of logical relations, which describe how these terms relate to each other.

Ontologies allow creating an effective information exchange system. The main task is not to collect disparate information, but structured, formal data to solve real business and economic challenges. The main purpose of information exchange system is to make information accessible and reusable across the whole system. Due to this fact that information, which is not described and not structured, eventually becoming worthless.

In contrast, information, which allows automated distribution and exchange generates added value. The entire above problem is solved in the system.

The ontological models of intellectual processing of the user data, data sets, resources, external systems for integration and sharing data were designed. It was necessary to develop a search algorithm (the agent) with the least loss of time; passing on the hierarchical structure of ontological models could qualitatively authenticated information.

As a basic standard description of the data set the SDMX Standards Version 2.0 was used so basic parameters of the statistical description of European repositories for automatic implementation of data integration between the repositories.

4. W-DMR presentation

Scientific data set Repositories are created for data storage, retrieval, correspondence and processing of data from different subject areas provide a valuable resource for researchers, teachers and students. The storage can help scientists to support their experiments in the field of data mining.

In our repository there are two users: beginner and expert. Each of them has own agent. The agent is used in different parts.

The beginner could: to find appropriate data set on a base of task description; try to add data and vice versa, choose concrete data mining domain using dataset.

The repository has Coordinator agent (manager). The user agents address to it and deliver tasks to beginner and expert agents. Each agent is a process that has a certain part of knowledge about the object and the opportunity to share this knowledge with other agents. The repository keeps Data Mining and Machine learning methods ontology models. We developed several ontology models - an ontology model of Data mining methods, an ontology model of the user, a model of the resource, which contain description about their structure and usage in W-DMR. Each ontology model has search agent. It receives information (tasks) from coordinator agent. Also dataset ontology model interacts with own dataset agent. And source ontology model interacts with coordinator agent.

The interaction between the ontological models is based on intelligent agents: coordinator agent, resource agent, search agent, user agent. The agent approach has been implemented by multi-technology Jadex. We used intelligent software agents. This is a new class of software systems, which acts either on behalf of the user, or on behalf of the system. They are, in fact, a new level of abstraction, different from the usual abstract type - classes, methods and functions. For practical implementation of these agents Jade offers to programmer-designer of agent systems the following possibilities: FIPA-compliant Agent Platform, which includes system agents AMS, ACC and DF; Multiple Domains support – DF agents and so on.

As a result, intelligent search agent interacts with the ontological model and the user is getting an expert answer to his query.

Search algorithm is based on ontological models and ranking results. Implementation of access to the intelligent agents based on web-services, implemented on the basis of programming paradigms such as dependency injection and control inversion. There were a variety of access through the implementation of web-projects and implemented a more flexible, robust architecture. During the work we analyzed agent-oriented approach methods and put the template design.

During the work we offered not only general model of the search module, but its detailed architecture and implementation. Architecture of intelligent search agent was created. The design patterns in agent-oriented approach were developed and.

5. System program model: Deployment & Implementation

This chapter presents the overall structure of the system. It describes all levels of the system.

Method, which implemented in the practical realization in this work, can be well characterized as Web 3.0. That suggests to researchers, teachers, students and other categories of interested users to download their statistical datasets including their meta description, to view already downloaded ones and to give advice to others regarding dataset.

Such users work is human strategy of «manager of knowledge», which leads the users to the desired results. The conventional technique to organize the process of information searching in databases provides for personal request of user via Internet to Web Data Mining Repository server with request a summary of the responses result and its treatment. Performance, in general, of routine operations may take the experts a lot of time. In this regard becomes acute the problem of development of multi-agent system to automate the process of the queries in the information system, which has to assume much of the routine operations in database systems. The overall structure of the system can be represented in Fig. 1.

The system consists of the presentation level, service level, agent's level and database.

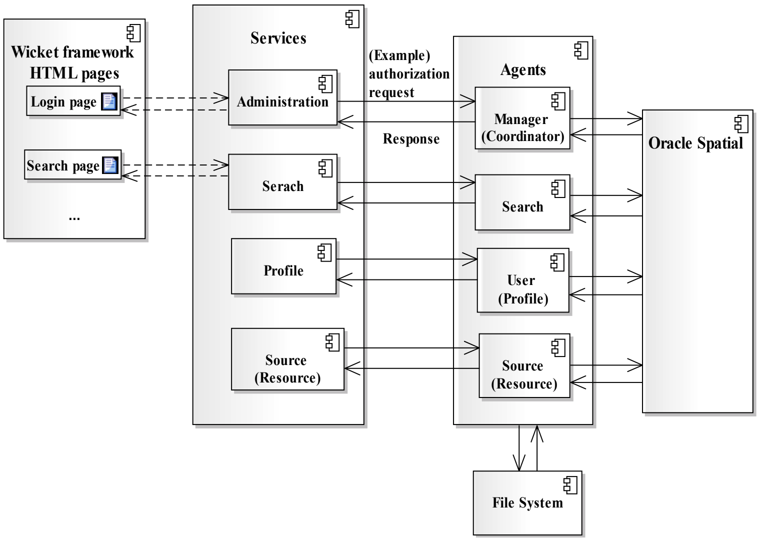

Fig. 1. The explanation of overall system

5. 1. Presentation level

All the Web part (Web pages) is a presentation level of the system. Presentation level was built using html-pages and Wicket Framework. Wicket is open software based on Web components. The pages divide into Markup files and code. Code is written on the Java language, an excellent support for localization and styles to pages, no xml-file configuration, easy integration with Java security. . Net programmers can easily compare it with ASP.NET pages. Of course now there are many frameworks for developing web applications but most frameworks have weaknesses in supporting the state of server components page. Wicket makes this support easy and transparent. Wicket operates independently as server components pages. Programmers do not need to personally use the Http Session object wrappers or similar storage condition. This is one of the of Wicket goals. Wicket pages scheme is shown in Fig. 2.

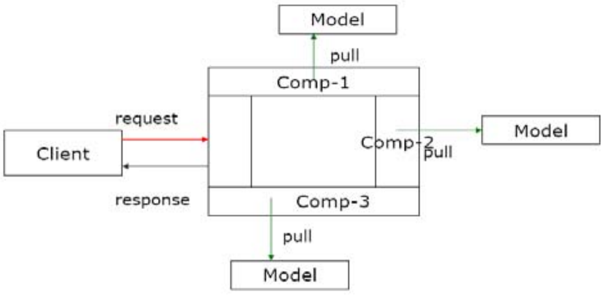

Fig. 2. Wicket pages work scheme

One of the part of general data and metadata exchange repository system was to create web-pages using program container Tomcat 5.5.

The Wicket page CreateDatasetPage.html of the system. It is provided that using the multi-agent developed client part the user forms a query to a distributed information system. This request passes through the all system levels to the server database.

Also in this section describes dataset part of the system. It may be considered the page to download data sets in the system, review of all data sets, view of detailed information about the datasets, edit metadata datasets. Fig. 3 shows diagram of presentation level class and service resource class.

The package ua.kture.dmr.common.beans.dataset has classes DataSet, DataSetFile, Judge. Each is an objective representation of metadata ontology resource. All classes of the system operate the objects of these classes.

Package ua.kture.dmr.agents.dataset provides classes plans agents:

– InsertDatasetPlan – performs plan insert_dataset, adds a new dataset to the repository;

– ReadAllDatasetPlan – performs plan read_all_dataset, reads all data samples from the repository;

– ReadDatasetsBySlotPlan – performs plan read_dataset_by_slot, reads all data samples that match the query from the repository;

– AppraisementDatasetPlan – performs plan appraisement_dataset, adds a specific set of assessment data, adds comment data set;

– UpdateDatasetPlan – performs plan update_dataset, updates data samples;

– InsertDatasetFile – performs plan insert_dataset_file, adds the files of statistical data sets.

The multi-server part of system is implemented in Java. This ResourceAgent provides communication interface with other agents and repository server system. This four-level repository architecture provides the opportunity to interact with developed repositories of services that are very urgent practice now and at the level of agents i.e. the interaction between agents of different systems is possible.

Package ua.kture.dmr.jwsx.ui.pages contains AbstractPage class, which is basic to all pages of the system. It creates a menu for each site page.

Package ua.kture.dmr.jwsx.ui.pages.dataset contains website pages CreateDatasetPage, DataSetListPage, DataSetDetailsPage, UpdatedataSetPage, which are inherited from AbstractPage base class. This package provides an interface to the data sets.

Package ua.kture.dmr.jwsx.wsimpl allows ResourceServiceImpl class provides implementation for queries to the source agent. ResourceServiceImpl class inherits from the AbstractAgentWebService class, which is the base for all system services and realized the ResourceService interface.

Features of class are listed below: insertDataSet (DataSet dataset) throws Exception; getDataSet (SessionInfo sessionInfo, String title) throws Exception; getAllDataSets (SessionInfo sessionInfo) throws Exception; getDataSetsBySlot (SessionInfo sessionInfo, String slotName, String slotValue) throws Exception; insertDataSetFile (DataSetFile datasetFile) throws Exception; setDataSetMarkComment (Judge judge) throws Exception; updateDataSet (DataSet dataset) throws Exception.

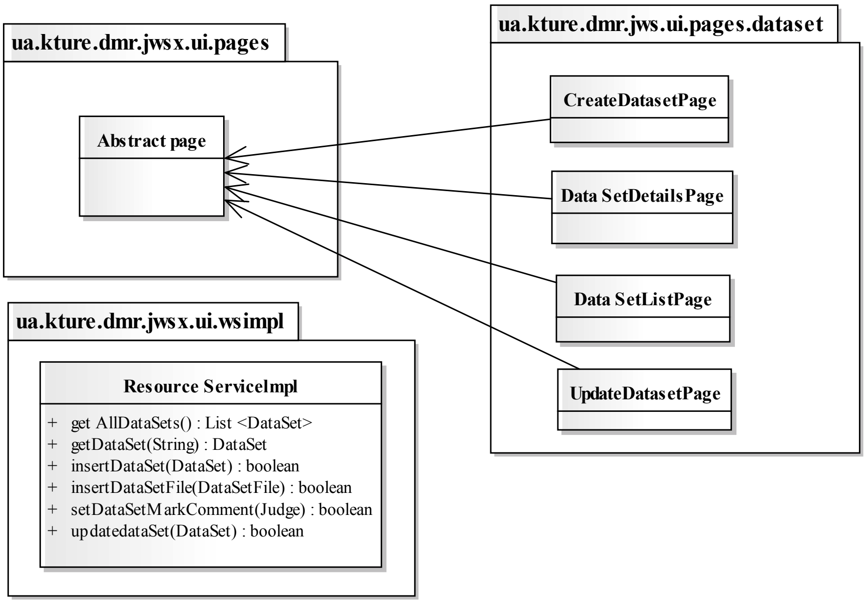
ua.kture.dmr.jwsx.ui.pages
Abstract page
ua.kture.dmr.jws.ui.pages.dataset
CreateDatasetPage
Data SetDetailsPage
Data SetListPage
UpdateDatasetPage
ua.kture.dmr.jwsx.ui.wsimpl
Resource ServiceImpl
+ get AllDataSets() : List <DataSet>
+ getDataSet(String) : DataSet
+ insertDataSet(DataSet) : boolean
+ insertDataSetFile(DataSetFile) : boolean
+ setDataSetMarkComment(Judge) : boolean
+ updatedataSet(DataSet) : boolean

Fig. 3. The presentation level and service level diagram class

5. 2. Service level

Web Data Mining Repository is a service-oriented architecture that meets the principles of multiple usages of the functional elements, eliminate duplication of functionality in the software, unification typical operating processes to ensure the operating model of centralized processes and functional organization based on the industrial platform integration. Components of the program can be distributed on different nodes of the network and offered as independent, weakly connected, which can follow service applications.

A developed software system is implemented as a set of Web services. It integrates using SOAP and WSDL. Interface of program components provides encapsulation of implementation details of specific component from other components. Thus, this architecture provides a flexible and elegant way to combine and reuse of components. In order to process the requests from users with Web interface was developed a multi-system. Web page requests directed to the web services system, which in their turn send requests to agents. The system supports four Web services: Administration Service; Search Service; Profile Service; Source Service. It was implemented the xfire solution To develop Web services. Xfire is a free solution that solves the problem of interoperability, implementation of various problems of industrial standards. Developers of distributed applications this is the easiest mechanism for implementing of remote requests. ResourceService service has a method that fully covers the functionality of the source agent.

5. 3. Agent subsystem

In the system Web Data Mining Repository all agents, which multi-system includes, belong to one of the following types:

– manager agent, running on the server and coordinates the work of users;

– user agent that performs the interaction with users;

– resource agent, responsible for datasets operations;

– agent search, performing the information search.

Thus, even if agents are placing on different servers, it will be possible to interact with queries from users. The multi-server system includes agents ManagerAgent, ProfileAgent, ResourceAgent, SearchAgent Messaging between agents based on the HTTP protocol and work with the database is via JDBC one.

5. 4. Work database level

To store ontological models there was used the database management system Oracle 10g, namely a new option Oracle Spatial. Each ontological model is designed for RDF DATA MODEL in Oracle Spatial. Thus we get three models: Users; Datasets; Methods.

DBMS Oracle Database 10g was the first large-scale project to implement storing ontologies in spatial form. Oracle Spatial is DBMS Oracle Database 10g technology which includes additional features for handling spatial data to support spatial services, various programs for processing or to provide information on the location of objects and other information systems. DBMS support includes Oracle 10g RDF / RDFS, allowing developers to use the platform to take advantage of semantic data. Application developers can add value to data and metadata, defining new sets of conditions and relations between them. This set of terms (ontology) is more suitable for query and analysis based on the semantic approach than conventional datasets. Otology datasets often contain millions of data elements and relations between them. It can be grouped in triplets using the new RDF data model. Oracle admits triplets billion expansion to meet the requirements of most applications.

How to store RDF in Oracle Spatial 10g: RDF data is stored as directed logical graph; Subjects and objects are displayed as nodes and predicates as relations, in which the subject is an initial node and final is object; Relationships are a complete RDF triplets; Oracle Spatial RDF data model; RDF data model supports three types of database objects: a model (RDF graph consisting of a set of triplets), base of rules (set of rules), the index rules (aimed RDF graph). To implement the semantic query is used SDO_RDF_MATCH operator;

The main advantages of Oracle Spatial 10g using are: Support for decentralized data management; Support of all RDF data types; SQL search and recovery of RDF models; Making queries to the RDF model, using the circuit graph; A query RDF (SPARQL) with other operators in SQL; A logical conclusion based on RDFS (RDF schema) rules; The logical conclusion based on policies defined by the annex.

RDF Model is stored as a graph: nodes – URI objects, certain set of links between nodes, W3C RDF Schema recommendation describes the dictionary, is applied to describe other dictionaries.

RDF documents are stored as a triplet (subject, property, and object) and use the reduction to represent namespace. A free library of Jena 2.0 is used for interaction with the database agents.

6. Conclusions

For ontological models interaction and implementation of search algorithms was developed a set of general intelligent agent models. They can be used as a mechanism for displaying information on the ontological models, as well as a mechanism for user interaction with the system. This set of general models include model for integrating intelligent agents with web systems, a model of intelligent search agent, and model for relationship between agents. The user of the developed intelligent data and web data mining repository is able to make formal description of the user's problem domain (filling in the necessary fields in the ontology model) and formal description of the dataset which is need for specific tasks.

The result of applying the multi-agent approach for creating such system is the ability to perform a simple search for users regardless of user type; to search by different criteria for authorized users; to provide popular data sets; to perform a search taking into account the personal needs of the user; to provide user relevant queries information; to keep statistics of requests and, if necessary, provide this information; to remember the successful search results.

Here is used the cross platform programming language Java, multi-agent platform Jadex, database server Oracle Spatial 10 g.

The complete development of a repository will allow to solve the problem of data use for beginners, will allow all scientists to exchange the descriptive part of files in different application areas. Adding files by various scientists it will not be necessary to fill in formally all fields to add the files. It will be enough to give files description and the agent will automatically add it in appropriate section, and further will find it for user.

References

1. Asuncion, A. UCI Machine Learning Repository [Electronic resource] / A. Asuncion, D. J. Newman. – University of California, School of Information and Computer Science, Irvine, CA, 2007. – Available at: http://www.ics.uci.edu/~mlearn/MLRepository.html

2. Blake, C. L. UCI repository of machine learning databases [Electronic resource] / C. L. Blake, C. J. Merz // Available at: http://www.ics.uci.edu/~mlearn/ML - Repository.html

3. Pearson, S. An Adaptive Privacy Management System For Data Repositories [Electronic resource] / S. Pearson, M. Mont, P. Bramhall. – Trusted Systems Laboratory, Hewlett-Packard Laboratories, Bristol, UK, 2004. – Available at: http://www.hpl.hp.com/techreports/2004/HPL-2004-211.pdf

4. Cunningham, K. An open repository and analysis tools for fine-grained longitudinal learner data [Electronic resource] / K. Cunningham, R. Kenneth, Koedinger, A. Skogsholm, B. Leber. – Human Computer Interaction Institute, Carnegie Mellon University, 2008. – Available at: http://www.educationaldatamining.org/EDM2008/uploads/proc/16_Koedinger_45.pdf

5. Xie, T. P. JMAPO: mining API usages from open source repositories [Electronic resource] / T. Xie, J. Pei. – Proceedings of the International Workshop on Mining Software Repositories (MSR '06), ACM Press, New York, 2006. – P. 54–57. – Available at: http://people.engr.ncsu.edu/txie/publications/msr06-mapo.pdf

6. Zimmermann, T. Knowledge Collaboration by Mining Software Repositories [Electronic resource] / T. Zimmermann. – Saarland University, Saarbrucken, Germany, 2006. – Available at: http://thomas-zimmermann.com/publications/files/zimmermann-kcsd-2006.pdf

References

1. Asuncion, A., Newman, D. J. (2007). UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA. Available at: http://www.ics.uci.edu/~mlearn/MLRepository.html

2. Blake, C. L., Mer, C. J. (2001). UCI repository of machine learning databases. Available at: http://www.ics.uci.edu /~mlearn/ ML - Repository.html

3. Pearson, S., Mont, M., Bramhall, P. (2004). An Adaptive Privacy Management System For Data Repositories. Trusted Systems Laboratory, Hewlett-Packard Laboratories, Bristol, UK. Available at: http://www.hpl.hp.com/techreports/2004/HPL-2004-211.pdf

4. Cunningham, K., Kenneth, R., Koedinger, Skogsholm, A., Leber, B. (2008). An open repository and analysis tools for fine-grained longitudinal learner data. Human Computer Interaction Institute, Carnegie Mellon University. Available at: http://www.educationaldatamining.org/EDM2008/uploads/proc/16_Koedinger_45.pdf

5. Xie, T., Pei, J. (2006). JMAPO: mining API usages from open source repositories. In: Proceedings of the International Workshop on Mining Software Repositories (MSR '06), ACM Press, New York, 54–57. Available at: http://people.engr.ncsu.edu/txie/publications/msr06-mapo.pdf

6. Zimmermann, T. (2006). Knowledge Collaboration by Mining Software Repositories. Saarland University, Saarbrucken, Germany. Available at: http://thomas-zimmermann.com/publications/ files/zimmermann-kcsd-2006.pdf

Рекомендовано до публікації д-р техн. наук Шамшм Б. В.
Дата надходження рукопису 25.12.2014

Tetyana Shatovska, Associate Professor, Department of Software Engineering, Kharkiv National University of Radioelectronics, Lenina, 16, Kharkov, Ukraine, 61166
E-mail: shatovska@gmail.com