

An Alternative Approach for Multidimensional Data Communication

Ivanka Valova

Institute of Control and Systems Research, 1113 Sofia

E-mail: vania@icsr.bas.bg

Abstract: *Subjects of analysis in this article are the software agents and brokers and its use regarding the formation of a communication infrastructure in the systems for Online Analytical Processing of multidimensional data. Following the rapid development of the software technologies, here are indicated the occurred changes and an additional explanation is given on the use of the software products, discussed in our previous articles related with the same issues. A special attention is paid to the infocubes, used for online analytical processing of data.*

Keywords: *multidimensional data, infocubes, Online Analytical Processing, communication infrastructure.*

1. Introduction

The rapid development of the information and computer technologies places new challenges before the researchers and the software developers. The presence of various software products sets as priority the issue about the communication between the various program modules. In our previous articles [3] we discussed the development of communication infrastructure for transfer of logic and data by use of Java, XML, ACL, KIF. In this article we are developing further this issue and our attention is focused on changes occurred during the usage of the relevant software products and mainly we turn our attention to the extraction and search of data, which is organized in form of infocubes, in accordance with the multidimensional paradigm used in Online Analytical Processing (OLAP) systems.

The proposed architecture is enabling the distribution of the processes among several computers. In case of error occurrence, the broker of queries is capable automatically to restart the processing.

The *software broker* (or *mediator of queries*) is arbiter in management of distributed objects. Each query of a customer, initiated by Graphical User Interface (GUI) is accepted by the broker, which activates process (agent) in a separate memory block. Each agent may take care for the execution of the query and for receiving of results. The broker may simultaneously manage multiple processes by optimizing the occupied memory [1].

The use of various types of agents for the formation of the communication infrastructure and its realization is described in [3]. In this article, we will refer to it as software agents.

The *software agents* are computer programs, which autonomously carry out the standard functions, like automatic search and extraction of information and reporting of exceptions.

The article has the following structure. In Section 2 we describe a simplified version for realization of GUI by using Java. Description of multidimensional model and the manners for implementation of the relevant analysis you will find in Chapter 3. The article ends with conclusions.

2. Realization of GUI

We show in Fig. 1 the summarized model for communication between various program modules. In this item, we will discuss the program realization of one simplified GUI (Fig. 2).

We define a *class SalesFrame*, which in the general case appears on the following manner, shown in Table 1.

From dropdown menu named “Groups products” is selected certain group of products, which will be analyzed. The information, which is visualized represents the relevant sales of the selected group of products. The use of *JavaScript* for access through Internet for the relevant realization is necessary. JVM from the Sun web page should be additionally installed for Windows XP.

3. Multidimensional model of distributed data and analysis of multidimensional models

In article [3] the data source were various types of files, Excel spreadsheets, etc., while here the data is organized in infocubes, which determines the use of new program techniques and technologies.

The *Infocube* (abbr. the cube) may be presented as a limited space of Cartesian n-functionally dependent levels of accumulation ($L_i \in D_i$), to a set of cells in class (C_c)
$$c : L_1 \times \dots \times L_n \rightarrow C_c;$$

Each cell in this cube represents an intersection point of different types *D-structures* – D_i at exact defined level and participates in the determination of quantity indicators of the fact. Measures evaluate attributes of fact [2].

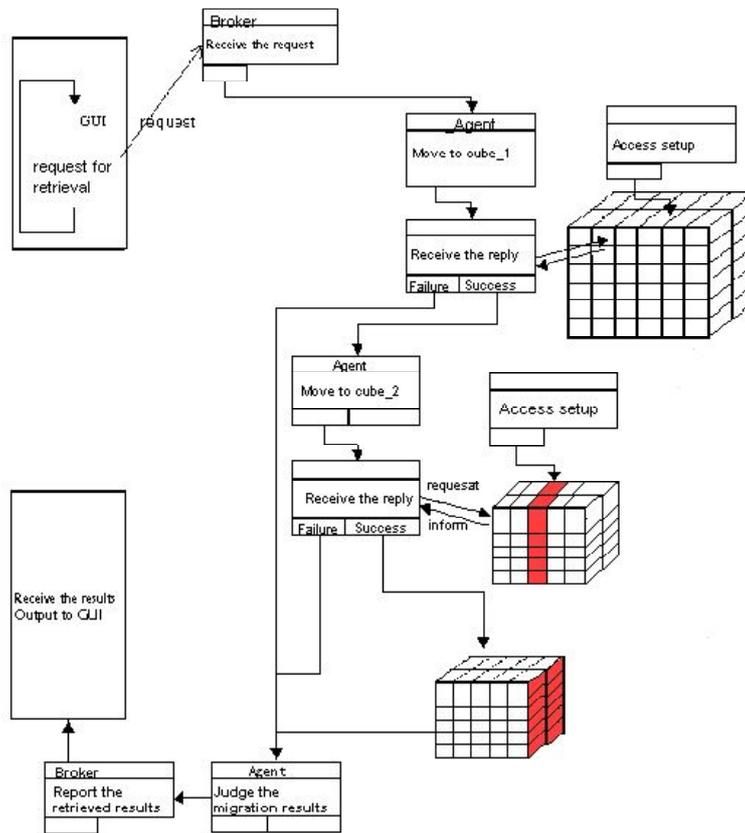


Fig. 1. Access distributed multidimensional data using XML/Java

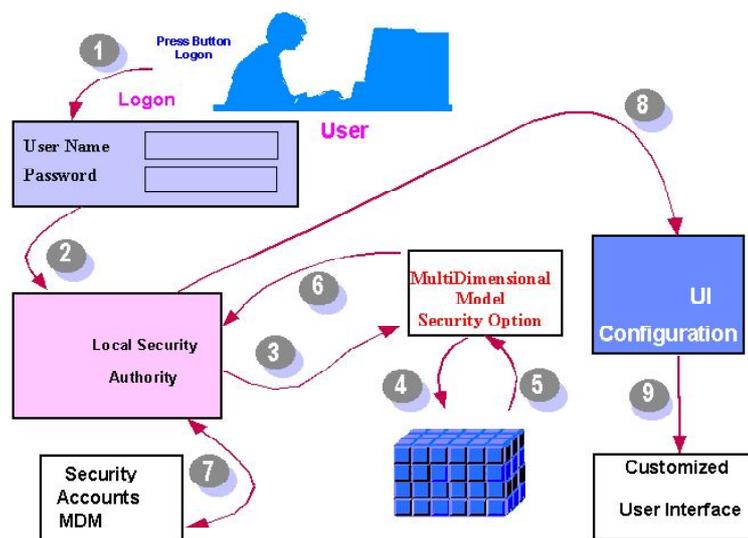


Fig. 2. Realization of simultaneous access to the network resources and to MDM

Table 1

<pre>public class SalesFrame extends Frame{ boolean adjFlag = false;</pre>		
<p>L a b e l</p>	<pre>Products; Gamount; Date; Iamount; GValue; DValue; IValue;</pre>	<pre>public void addNotify(){ Dimension d = getSize(); super.addNotify(); if (adjFlag) return; setSize(getInsets().left + getInsets().right + d.width, getInsets().top + getInsets().bottom + d.height); Component components[] = getComponents(); for (int i = 0; i < components.length; i++){ Point p = components[i].getLocation(); p.translate(getInsets().left, getInsets().top); components[i].setLocation(p); } adjFlag = true; }</pre>
	<pre>public SalesFrame(){ setLayout(null); setVisible(false); setSize(getInsets().left + getInsets().right + 375, getInsets().top + getInsets().bottom + 200); setBackground(new Color(8421504)); setTitle("Analysis of MDM"); addWindowListener(new WinAdapter()); ProductGroups = new Choice(); ProductGroups.setBounds(getInsets().left + 144, getInsets().top + 24, 204, 40); ProductGroups.setBackground(new Color(16777215)); ProductGroups.addItem("PrGroupA 1010"); try {ProductGroups.select(-1); } catch (IllegalArgumentException e) { } add(ProductGroups);</pre>	<pre>class WinAdapter extends WindowAdapter{ public void windowClosing(WindowEvent event){ Object object = event.getSource(); if(object == SalesFrame.this){ setVisible(false); dispose(); System.exit(0); } } }</pre>
	<pre>Products = new Label("Products"); Products.setBounds(getInsets().left+ 24, getInsets().top + 36, 100, 24); Products.setFont(new Font("Dialog", Font.PLAIN, 18)); add(Products); Date = new Label("Data (dm)"); Date.setBounds(getInsets().left + 24, getInsets().top + 110, 100, 24); Date.setFont(new Font("Dialog", Font.PLAIN, 18)); add(Date); DValue = new Label("000000", Label.RIGHT); DValue.setBounds(getInsets().left + 160, getInsets().top + 110, 180, 24); DValue.setFont(new Font("Dialog", Font.BOLD, 20)); DValue.setForeground(new Color(65280)); DValue.setBackground(new Color(0)); add(DValue);</pre>	<pre>public void setItemListener(ItemListener il){ ProductGroups.addItemListener(il); } public void setGamount(String s){ GValue.setText(s); } public void setDateValue(String s){ DValue.setText(s); } public void setIamount(String s){ IValue.setText(s); } public static void main(String args[]){ new SalesFrame().setVisible(true); }</pre>

The infocube may be designed by using object-oriented approach. We define Class Infocube on the following manner.

The class *Infocube* (*C*) is defined as a triad (*D*, M_{ic} , M_c) where:

1. *D* is set of classes *D*-structures.

2. M_{ic} is set of master operations for class *C* (for example the operations “new”, “delete”).

3. M_c is set of methods for class *Infocube*. M_c includes the methods- find objects, get_root_nodeid, get the first level, call_protokol, set selected node, raise event for netgraphic

It is necessary to define the access authorizations of each user. For that purpose is necessary to program additional module indicated on Fig. 2 as *Accesses setup*. In most software systems each user should register first with name and password to gain access to the network resources of the relevant computer network and then to register again to gain access to the multidimensional modules, which are part of the OLAP systems. This raise the necessity of development of program, which will allow single registration of the users to gain access to all program modules without violating the security of the accessed data Fig. 2. The development of similar software is discussed in article [4].

After the design of the infocube, the access and the extraction of the data should be carried out by using the software agents.

The infocube on Fig. 3 contains data on the following *D*-structures – Time, Product, Region, Company code, Profit, Department, Customers group, etc. From the available data should be selected only those, which represents interest for the particular user. On Fig. 3 are selected – Time, Product, Region. The resulting cube may be analyzed for obtaining of more detailed information. Arrow number 2 shows selection of data through software agent for product P2.

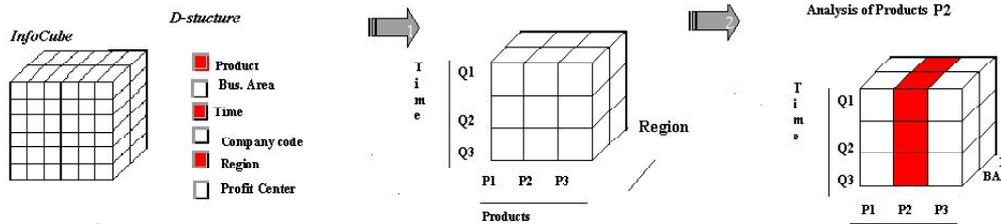


Fig. 3. Analysis of multidimensional models by using of software agents

By using the software agents, an access may be realized to more selective aspects of the data by use of filter values in any particular request. Further, it is possible to analyze the specific sectors and regions, which are present in the discussed multidimensional model.

Disadvantage of the proposed architecture is the high price for realization and the necessity of specially trained personnel for realization and development of the system.

Conclusion

In this paper, we presented an alternative approach for extraction, search and presentation of data from multidimensional models by using of software agents and brokers. The proposed communication infrastructure and the relevant software packages provide a solution of the issue for the relation between various program modules and may be further developed and used by software developers and research teams, working in field of the knowledge-based management systems and in field of integrated systems for digital progressing of signals.

The purpose of this paper is to focus the attention on changes occurred in result of the rapid development of the computer technologies. We presented here a multidimensional model, where the data is accessed and extracted by software agents.

RRRRRRRRRR

1. G r o u m p o s, P., A. S m i r n o v. Multi-Agent based virtual enterprise configuration management: general principles and basic technologies. – In: IFAC LSS'98 Large Scale Systems: Theory and Applications, Rio Patras, Greece, 1998.
2. V a l o v a, I., B. Z h e c h e v, V. V a l o v. Overview of research and software approaches for multidimensional data analysis. – Cybernetics and Information Technologies, Vol. 1, 2004, No 1.
3. V a l o v a, I., B. Z h e c h e v. MDM of distributed data and XML approach for communication. – In: Cybernetics and Systems'2002, Austrian Soc. for Cybernetic Studies, Vol. 2. R. Trappl, Ed. 890-893
4. V a l o v a, I. Central control of multiple user accounts in Windows NT and Oracle database. – In: PCaPAC 2000, October 9-12 2000, Hamburg, Germany.

Алтернативен подход за комуникация на многомерни данни

Иванка Валова

Институт по системи за управление и изследване, 1113 София

(Резюме)

В тази статия обект на анализ са софтуерните агенти и брокери и тяхното използване за оформяне на комуникационна инфраструктура в системите за аналитична обработка на многомерни данни в реално време. Като е разгледано бързото развитие на софтуерните технологии, са посочени настъпилите промени и е дообяснено използването на софтуерните продукти, дискутирани в предишни наши статии, свързани с разглежданата тематика. Специално внимание е отделено на инфокубовете като място за съхранение на многомерни данни, които ще бъдат използвани за аналитична обработка на данни в реално време.