Internet Traffic Analysis System Based on Data Mining and OLAP

Jovanka D. Cekic[1], Saša B. Spasic[2]

Abstract – **The application of Data Mining and OLAP technologies in the Internet traffic analysis is described in this paper. It presents one of the solutions for the analysis of log files and data provided over SNMP protocol based on Open Source tools, whose low price facilitates its employment in small and medium-size enterprises.**

Keywords – **Internet traffic analysis, data mining, OLAP**

I. Introduction

Internet traffic data analysis is a very important part of Internet engineering and management, because traffic data measure very complex characteristics of network traffic. The vastness of the Internet network topology, together with complex statistical properties of Internet traffic data and very large databases, make the analysis very challenging. In this paper we present a method of analyzing log files of different applications and devices on the network, which can contain very heterogeneous data – IP addresses, timestamps, actions taken, response, etc, and can result in very large database.

The success of analysis itself depends mainly on ability to analyze the log database in detail. It is important to explore the raw data, because relying only on summaries only could be inadequate. Therefore, we have chosen data mining and OLAP techniques to perform a complex and necessary analysis.

Data mining, also known as "knowledge-discovery in databases (KDD)**",** has been defined as "The science of extracting useful information from large data sets or databases" [1]. It usually uses computational techniques from statistics and pattern recognition and is usually used in relation to analysis of data.

Online Analytical processing (OLAP) is an approach to quickly providing the answer to complex database queries, and is used in business reporting for sales, marketing, management reporting, data mining and similar areas. [2]

II. Data Mining Using OLAP

OLAP technology enables efficient use of large databases for online analysis, providing quick responses to complex analytical queries.

OLAP uses multidimensional data model and data aggregation techniques to organize and summarize large amounts of data, so it can be viewed and analyzed using online analysis and graphical tools. OLAP systems provide speed and flexibility to support real time analysis. [3]

The most efficient way of organizing an OLAP database is a multidimensional Cube that is sometimes called a database of subtotals.

Multidimensional cubes are created from data in the data warehouse fact and dimension tables. A fact table contains the measurements or facts of business processes and foreign keys for the dimension tables. [4] Dimension tables contain the context (i.e. characteristics) of the measurements. Each dimension table contains data for one dimension. The Dimension Attributes are the various columns in a dimension table. Dimension tables indicate how the aggregations of relational data can be analyzed. Dimension data are hierarchically organized. Multidimensional structures in which cubes are stored are specially designed for rapid query response, and they can contain data summarized, copied or directly read from the data warehouse [5].

Storing data in OLAP cubes, instead of in relational tables, provides more efficient way of retrieving data for reporting purposes. The key concept that provides faster data retrieval from the cubes is data aggregation. It is much easier to retrieve data that are already associated, than to perform a complex relational database search.

OLAP servers are categorized according to the way of data storage: Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP), and Hybrid OLAP (HOLAP) [2].

In a MOLAP model cubes are stored in multidimensional database files. Data are stored on disk, in structures optimized for multidimensional access. The required schema contains a dimensional set of both base data and aggregations. Data access is very fast, and memory usage is very high.

In a ROLAP-model a multidimensional data cube is mapped to relational model, so that one cube is mapped to several relational tables. Special tables are created to hold the aggregation information.

HOLAP (Hybrid OLAP) combines these two models – multidimensional database files and relational tables.

III. Operations on the Cube

What gives the extra qualities to the analysis of the Cube, are operations on presented data. The user starts from one view of the data, choosing two dimensions for X and Y axis, and fact data for the table fields. The following operations are available:

1. Drill-down – a view of data more precisely grouped on the lower level of hierarchy.
2. Slice – segment selection on one dimension.
3. Dice – changing of chosen dimensions for the view

[1]Jovanka D. Cekic is with the University of Nis, Univerzitetski trg 2, 18000 Nis, Serbia and Montenegro, E-mail: jovanka@ni.ac.yu
[2]Sasa B. Spasic is with the "IRVAS" International, Nikole Pasica 32/4, 18000 Nis, Serbia and Montenegro E-mail: spaske@irvas.co.yu

IV. Architecture of an OLAP Based System

The Fig. 1. represents the architecture of a system based on OLAP. The transaction data repository contains data created by the application or the system that is being analyzed. These can, for example, be on-line sales data or, in this case, log files produced by the applications or devices in the computer network.

Data import procedure is a scheduled task (it is executed once per hour or day, or more often, depending on the data nature). It analyzes transactional data and writes them into the Cube, i.e. multidimensional database.

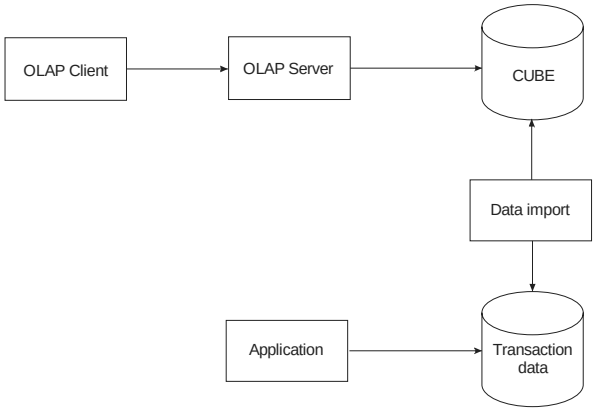

Fig 1. Typical OLAP system architecure

There is a large number of tools that can be used for implementing this kind of system. Microsoft offers a set of its own tools:

1. OLAP client – Excel pivot table
2. OLAP server – MS Analysis Server
3. Cube – MS Analysis Server
4. Transactional Data – MS SQL
5. Protocol between OLAP client and server - MS proprietary protocol using OLE DB provider for OLAP

Client and server can also be implemented using open source technologies:

1. OLAP client – Internet browser as a client, and Jpivot (jpivot.sf.net) as a Web application
2. OLAP Server – Mondrian [4]
3. Cube – any RDBMS can serve as a ROLAP repository (for example MySQL)
4. Transactional data – MySQL, log files etc.
5. Protocol – a custom protocol implemented using TCP/IP or XML/A

Also, there is a large number of very complex systems that implement data mining using OLAP, but their price as well as the price of their deployment (which often goes up to several hundreds thousand dollars) makes them unprofitable for application in the Internet traffic analysis.

The system presented on the Fig. 2 is based on ROLAP and open source technologies - Jpivot is used as an OLAP client and Mondrian as an OLAP server Therefore, it can be used for almost any type of analysis, because it provides a possibility of very fast data retrieval on one side and has a very low price on the other.

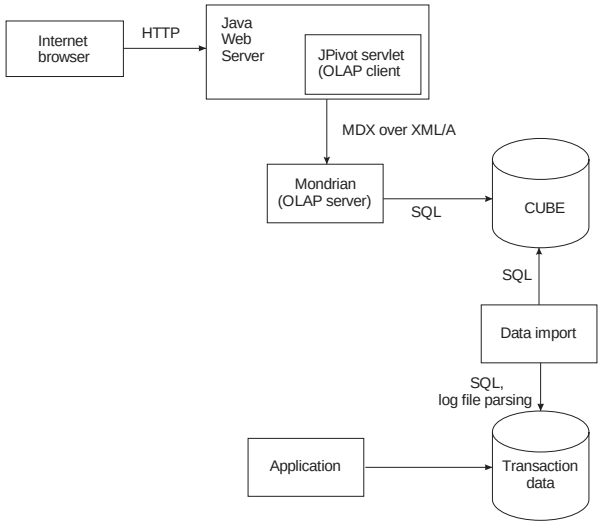

Fig 2. Applied OLAP system architecture

V. MDX – OLAP query language

MDX is an acronym for **M**ulti**d**imensional E**x**pressions. It is defined by Microsoft as "a syntax that supports the definition and manipulation of multidimensional objects and data" [3]. MDX is in many ways similar to SQL (Structured Query Language), but it is not an extension of the SQL. SQL cannot be used for so efficient an implementation of the features supplied by MDX.

Like SQL query, each MDX query requires a data request (the SELECT clause), a starting point (the FROM clause), and a filter (the WHERE clause). In that way specific portions of data can be easily extracted for analysis. MDX also contains a large set of functions for the data manipulation, as well as the ability to be extended with user-defined functions.

MDX also provides data definition language (DDL) syntax for managing data structures. It has commands for creating (and deleting) cubes, dimensions, measures, and their subordinate objects.

The purpose of Multidimensional Expressions (MDX) is to make data access easier and more intuitive by using multiple dimensions. MDX returns a subset of multidimensional data from cubes.

VI. XML/A – Web service for MDX queries

XML for Analysis, or XML/A [6], [7], formerly called Thin OLAP, is designed specifically for standardizing the protocol (communication between a client application and a data provider) over the HTTP. It implements the protocol by use of SOAP Web services. This standard has been supported by Microsoft, and the next releases of Microsoft Excel are expected to have a direct support for XML/A access to OLAP server. It supports the exchange of analytical data between clients and servers on any platform and with any language.

VII. ROLAP TOOLS

As we have chosen the ROLAP model and the open source tools for the Internet traffic analysis, the role of OLAP client component is taken over by Jpivot. Jpivot is a Java Web application, which runs on Java Servlet Container (for example Tomcat), and is by the way of working very similar to Excel pivot tables. Mondrian, which is also Java Web application, takes the role of an OLAP server, which provides XML/A Web service as a communication interface, and also a special protocol that runs over TCP/IP.

There are two more components missing for the completion of the system:

1. The component that creates and prepares ROLAP tables for the data, configuration files for Mondrian and Jpivot.

2. The component for incremental import of the analyzed data into the ROLAP tables.

VIII. DATA COLLECTING PROGRAM FOR ANALYSIS FROM SNMP AND LOG FILES

The system for analysis has to be applicable to arbitrary log files. Therefore, it is necessary to describe the contents of a log and the log files rotation strategy by means of an additional configuration file.

The part of the application related to data import part on the Fig. 2. contains two modules:

1. The first module analyses configuration files that contain the description of the repository with transactional data (log files, their position on disk and format, SNMP - Simple Network Management Protocol, data source, calculated data) and creates ROLAP mapping of the Cube. Additionally, this module creates a configuration for Mondrian and Jpivot based on configuration files.

2. The second module, which is executed periodically, analyzes log files, i.e. SNMP data sources and incrementally inserts data into the Cube.

IX. MODULE FOR ROLAP TABLES CREATION

Transactional data shown on the Fig. 1, which are monitored in a computer network, can be divided into two sets:

1. Log files with transactions
2. Traffic load on network-links

When a log file is the data source for the first module, for example a Web server log file, each data, except load (which is fact, or in Mondrian terminology measure column of fact table), is being declared a dimension. Dimension hierarchies are organized on that occasion, too:

1. Dates are grouped by month, and next on the higher level by year.

2. HTTP status codes can be grouped in the way described in HTTP specification [8] (1xx – informational, 2xx – success, 3xx – redirection, 4xx – client side errors and 5xx – server side errors).

3. URLs are grouped according to the configuration file.

4. IP address of the client grouped by arbitrary criteria (domain, country, geographic position etc).

In the configuration file, it is possible to define additional dimensions according to an arbitrary HTTP header or cookie. In that way it is possible to obtain the following data:

1. The data about browser, that are classified according to the browser type (Internet Explorer, Mozilla, etc.) on the higher, and according to the browser version on the lower level.

2. The data about users, which are classified by the application they are using (for example unregistered visitors, registered visitors, application administrators, etc).

3. The data about sessions, which are recognized on the basis of cookies, URLs or combined (that is very common with Java Web applications).

4. Hosts with virtual hosting can be classified according to different criteria.

If the data sources are SNMP queries, the first module recognizes the following data types as dimensions:

1. Octet String
2. Display String
3. Object Identifier
4. IP Address
5. Physical Address
6. Time Ticks

The following data types are recognized as a measure:

1. Integer
2. Counter
3. Gauge

Sometimes, the data for measure types represent dimensions, and they can be configured in that manner.
SNMP Sequences can be analyzed by treating their components separately, as a measure, dimension or a sequence, what is also defined in the configuration file.

The configuration file provides a possibility to define Calculated Members (with Mondrian) using a formula based on Measure data. Additionally, this module creates tables for storing data about the last import, provided by the module for update. That provides this module the possibility to operate incrementally.

X. MODULE FOR INCREMENTAL DATA UPDATE

In the case that data source is a log file, the module for incremental data update extracts a new part of the log, according to the existing data related to the last update and the strategy of log files rotation, and copies the data into the fact

table, updating in that way the dimension tables and inserting missing coordinates.

If the fact table contains calculated data, the module makes calculations based on the configured formulae, and stores the results in corresponding fact table columns. In case of SNMP data source the queries are sent by means of SNMP 2.x or 3.x protocol, and the further data processing is identical as with log files.

At the end of update the data about module progress are recorded, so that the next update would be incremental.

XI. Conclusion

ROLAP model has its limitations in comparison with MOLAP, because a model for generating indexes for relational tables cannot be efficiently used for indexing a database created in this manner. Also, a similar problem occurs when two dimensions are represented on a higher level of hierarchy – sometimes, in order to get a query results, it is necessary to analyze very large number of records from the fact table.

Mondrian has a caching mechanism that enables it to exceed these disadvantages, but it is not that efficient as with MOLAP.

The main advantage of this system is that OLAP Analysis offers more possibilities in comparison with common static analyses regarding data search (for example the impact of one separate component on the whole system or the web application parts efficiency), which may be incomplete in the classic analysis.

The low price of the software being used and open source components allows this solution to be used in smaller networks and enterprises where the internet traffic itself is of no primary importance. The components are, nevertheless, fast enough to make the analysis interactive.

References

[1] D. Hand, H. Mannila, P. Smyth: *Principles of Data Mining*. MIT Press, Cambridge, MA, 2001.

[2] E.F. Codd, S.B. Codd, C.T. Salley: “Providing OLAP (on-line analytical processing) to User-Analysts: An IT Mandate”, *Technical Report*, 1993 http://www.hyperion.com/products/whitepapers/

[3] Microsoft, *OLE DB Programmer's Reference*, Chapter 26, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/oledb/htm/olapmdxgrammar.asp

[4] Julian Hyde, *How to design a Mondrian schema*, http://mondrian.sourceforge.net/schema.html

[5] Ming-Syan Chen, Jiawei Hah, Philip S. Yu: “Data Mining: An Overview from a Database Perspective”, *Ieee Trans. On Knowledge And Data Engineering*, 1997 http://cgi.di.uoa.gr/~pms510/Papers/han.pdf

[6] Robin Grosset, *The case for XML for Analysis*, http://www.xmla.org/download.asp?id=83

[7] Microsoft, Hyperion, *XML for Analysis Specification, Version 1.0*, http://www.xmla.org/download.asp?id=2

[8] World Wide Web Consortium, Hypertext Transfer Protocol - HTTP/1.1, 1999, http://www.ietf.org/rfc/rfc2616.txt