

DOI: 10.18721/JCSTCS.11402
 УДК 004.655:004.514;004.657

АЛГОРИТМЫ ДИНАМИЧЕСКОЙ ГЕНЕРАЦИИ MDX-ЗАПРОСОВ К МНОГОМЕРНЫМ OLAP-КУБАМ

С.Г. Попов, А.А. Лисенкова

Санкт-Петербургский политехнический университет Петра Великого,
 Санкт-Петербург, Российская Федерация

При работе с многомерными базами данных возникает проблема длительного времени доступа к данным. Большинство серверов, поддерживающих технологию OLAP, хранят данные на дисках, тем самым увеличивая время доступа к требуемым данным. Альтернативным способом хранения является использование оперативной памяти, которое, однако, накладывает ограничение на объем памяти. Сервер Palo организован по принципам MOLAP и в качестве хранилища данных использует оперативную память. На данный момент доступ к данным и метаданным сервера осуществляется только с помощью атомарных веб-запросов. Применение MDX расширяет функционал доступа к серверу, а также унифицирует протокол обращения к данным. Результатом работы является подсистема генерации MDX-запроса на выборку данных из многомерных кубов OLAP-сервера Palo с интерактивным графическим интерфейсом. В проекте решены задачи анализа программных средств для работы с MDX-запросами, исследования синтаксиса MDX-запроса на выборку данных, разработки алгоритмов извлечения данных и метаданных из OLAP-кубов, а также реализации подсистемы генерации MDX-запроса на выборку данных из OLAP-куба с интерактивным графическим интерфейсом. Результаты могут использоваться для построения подсистем удаленного доступа к OLAP-серверу Palo.

Ключевые слова: OLAP, MOLAP, MDX, PALO, многомерные кубы, базы данных, метаданные, запросы, динамическая генерация, хранилище данных.

Ссылка при цитировании: Попов С.Г., Лисенкова А.А. Алгоритмы динамической генерации MDX-запросов к многомерным OLAP-кубам // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 21–35. DOI: 10.18721/JCSTCS.11402.

ALGORITHMS FOR MDX-QUERY GENERATION IN MULTIDIMENSIONAL OLAP-CUBES

S.G. Popov, A.A. Lisenkova

Peter the Great St. Petersburg Polytechnic University,
 St. Petersburg, Russian Federation

When working with multidimensional databases, there is a problem of long time access to data. Most servers that support OLAP-technology store data on disks, thus increasing the access time to the required data. An alternative method to store data is using RAM; this, however, imposes a limitation on the amount of memory. The Palo server is organized in accordance with MOLAP principles, and uses random access memory as data storage. Access to the data and metadata of the server is currently provided only through atomic Web requests. Use of MDX extends the functionality of access to the server, and also unifies the protocol of access to data. The result of the study is the subsystem generating MDX-queries for data retrieval from multidimensional cubes of OLAP-server Palo with interactive graphical interface. The project solved the tasks of analyzing software

tools for working with MDX-queries, examining the syntax of an MDX-query for data retrieval, developing algorithms for extracting data and metadata from OLAP-cubes, and implementing the MDX-query generation sub-system for retrieving data from an OLAP-cube with interactive graphical interface. The results can be used to build remote access subsystems to the Palo OLAP-server.

Keywords: OLAP, DBMS, MOLAP, PALO multidimensional cubes, metadata, MDX, queries, dynamic generation, data storage.

Citation: Popov S.G., Lisenkova A.A. Algorithms for MDX-query generation in multidimensional OLAP-cubes. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 21–35. DOI: 10.18721/JCSTCS.11402.

Введение

Технологии OLAP предоставления данных обеспечивают хранение и предоставление данных для комплексного анализа, состоящее в подготовке обобщенных данных больших массивов, структурированных на основе многомерного принципа. Совокупность естественных и интуитивно понятных моделей, организованных в виде многомерных кубов, обеспечивают чтение данных прикладными программами пользователей. Это позволяет использовать OLAP-системы в области прикладных программ для подготовки данных для бизнес-отчетности, предполагающих представление данных в различных срезах и разных уровнях иерархии, таких как отчеты по продажам, различные формы бюджетов государственных и коммерческих организаций, в том числе и в таких формах анализа данных, как интерактивное прогнозирование трендов [11, 12, 15], что требует быстрого доступа к данным OLAP-систем.

Доступ к данным может осуществляться при помощи нескольких видов интерфейсов, к которым относится доступ к отдельным элементам многомерных кубов при помощи интерфейса MOLAP, реализованным, например, через web-запросы, или декларативных языков управления многомерными данными MDX. Различные OLAP-системы имеют свои программные интерфейсы для реализации запросов, и не во всех информационных системах существует возможность использования языка запросов MDX. В случае реализации типовых решений, основанных на выполнении запросов из специально разработанного программного обеспечения,

реализация методов как при помощи web-запросов, так и при помощи MDX определяется программной реализацией и не представляет сложности для пользователя, т. к. сложность реализации скрыта за интерфейсом предоставления данных. Однако в случае, если требуется выполнить единичный аналитический запрос, требующий выборки данных из куба, перед конечным пользователем, чаще всего аналитиком, встает проблема получения данных. Она связана с формулировкой результата и передачи её для выполнения специалисту-программисту, который выполнит эти запросы в консольном приложении или путем написания фрагмента кода. Указанный путь долгов, т. к. требует участия нескольких специалистов и написания уникального программного кода для каждого запроса на получение данных. При этом аналитики, желающие получить данные из кубов, не будучи знакомыми с низкоуровневыми программными интерфейсами, знакомы с высокоуровневыми языками запросов, такими как MDX. Реализация нового приложения, обеспечивающего аналитика средствами доступа к данным куба средствами MDX-запросов, исключит трудоемкий процесс программирования запроса и ускорит получение данных. Реализация такого MDX-запроса возможна с использованием интерактивных графических форм, предоставляющих шаблоны запроса и требующие только дозаполнения данных о конкретных кубах. Применение интерактивных графических интерфейсов для доступа к многомерным OLAP-кубам позволяет расширить число пользователей, употребляющих технологию OLAP.

Цели и задачи исследования.

Схожие направления исследований

Цель настоящего исследования – разработка алгоритмов и реализация интерактивного графического интерфейса для динамической генерации MDX-запроса на выборку данных из многомерного OLAP-куба, располагающегося на MOLAP-сервере Palo, что предполагает выбор ограничений синтаксиса MDX-запроса на выборку, разработку алгоритма получения метаданных OLAP-куба для построения запроса на выборку, формирование интерактивной экранной формы для выбора атрибутов куба, разработку алгоритма формирования и выполнения MDX-запроса и исследование времени функционирования подсистемы генерации MDX-запроса на выборку данных из OLAP-куба с интерактивным графическим интерфейсом.

Исследование автоматической генерации запросов к OLAP-кубам из графических интерфейсов – активно развивающаяся область исследований. За последние пять лет основными направлениями исследований стали разработка языка обращения к многомерному пространству [8, 14], построение OLAP-кубов данных с помощью стандартных средств web-приложений [9], разработка технологий унифицированных языков доступа к данным куба [10, 13].

В работе [1] описано построение системы, работающей с многомерными аналитическими пространствами для обработки информации – OLAP-кубами данных, с помощью свободно распространяемых бесплатных инструментальных и программных средств для разработки web-приложений. Рассмотрены вопросы построения структуры БД, разработки системы кодирования и основанного на ней языка обращения к многомерному пространству, а также построения и проектирования транслятора, преобразующего выражения, написанные на языке обращений к многомерному пространству, в SQL-подобный язык запросов к БД. Использование описанного в этой статье подхода позволяет осуществлять разработку и построение OLAP-кубов данных

на стандартных средствах разработки web-приложений без привлечения дорогостоящих специализированных программных средств для работы с многомерными аналитическими пространствами, выраженными в форме OLAP-кубов данных.

В статье [2] изучена проблема общения оператора терминала с информационной системой на проблемно-ограниченном подмножестве естественного языка. Предложен механизм семантического анализатора, осуществляющего перевод запроса к информационной системе с естественного языка на SQL-подобный язык запросов среды IC: Предприятие. Приведена блок-схема самообучения семантического анализатора. Вся совокупность запросов на естественном языке ограничена двумя классами предложений: простые вопросительные и побудительные предложения. Приведены примеры, на которых подробно рассмотрены этапы преобразования запросов с естественного языка.

Представление дизайна и прототипа фреймворка для представления OLAP-операций в виде набора RESTful Web Services рассмотрено в [3]. Web Services были разработаны для загрузки куба, среза, drill-down и других операций.

Представление eLog – BI решения для eBilling для отслеживания управления документами, оптимизации и анализа организовано в работе [4]. Представляет простой и интуитивно-понятный метаязык для неопытных пользователей. Для оптимизации MDX-запросов создали алгоритм для оптимальной идентификации куба среди всех кубов, доступных в хранилище.

В статье [5] приведено описание реляционной OLAP-системы архивных метеоданных, созданной на базе разработанной ROLAP-Mining СПО системы. Ядром системы является СУБД MySQL. MDX-запросы визуализируются с помощью плагина Saiku Analytics. В системе используется технология интеллектуального анализа данных.

В работе [6] предложена OLAP-структура, основанная на web-технологии, которая была разработана из-за недостатков системы OLAP в рамках обычной структу-

ры клиент-сервер. Изучив фактические примеры, была разработана многомерная модель аналитической системы с применением методов связи между Office Web Components и сервером анализа. Кроме этого, реализованы многие операции OLAP-анализа для многомерных данных.

После представления временной модели данных, поддерживающей исторические измерения и ведение таблицы фактов, анонсирована трехуровневая архитектура, основанная на web-сервисах SOAP и XML, что позволило эффективно запрашивать данные из web-источников. В этой архитектуре XML-метаданные хранятся на сервере приложения в виде XML-документов, содержащих структуру хранилищ данных. Это позволяет разрешить доступ для запросов без использования хранилища. Результаты работы представлены в [7].

В [8] рассмотрен системный дизайн компилятора MDX с помощью метода ADD. Это означает, что системные требования, включая функциональные и качественные требования и ограничения, рассматриваются как драйверы в процессе проектирования. Конечная архитектура удовлетворяет не только функциональным требованиям, но и важным качествам, которыми должна обладать система.

Синтаксис MDX-запроса к многомерному OLAP-кубу

Реализуемый запрос на выборку данных из многомерных OLAP-кубов с помощью языка MDX представляет собой набор команд, который имеет следующий вид:

$$SELECT \left[\begin{array}{l} \langle SELECT \text{ query axis clause} \rangle \\ [, \langle SELECT \text{ query axis clause} \rangle \dots] \end{array} \right]$$

$$FROM [\langle SELECT \text{ subcube clause} \rangle]$$

$$[\langle SELECT \text{ slicer axis clause} \rangle]$$

Оси запроса определяют границы набора ячеек, который возвращается инструкцией многомерных выражений SELECT. Определение границ набора ячеек позволяет сократить количество возвращаемых данных.

Для определения осей запроса используется инструкция $\langle SELECT \text{ query axis clause} \rangle$, связывающая набор данных с определенной осью запроса. Каждое значение в этой инструкции определяет одну ось запроса. Таким образом, число осей в наборе соответствует числу значений $\langle SELECT \text{ query axis clause} \rangle$ в инструкции SELECT.

Для определения оси в инструкции $\langle SELECT \text{ query axis clause} \rangle$ используется следующий синтаксис:

$$\langle SELECT \text{ query axis clause} \rangle ::= [NON \text{ EMPTY}] \text{ Set Expression}$$

$$[[\langle SELECT \text{ dimension property list clause} \rangle]] [\langle HAVING \text{ clause} \rangle]$$

$$ON \{$$

$$\text{ Integer Expression } |$$

$$AXIS (\text{ Integer Expression }) |$$

$$\{ COLUMNS | ROWS | PAGES | SECTIONS | CHAPTERS \}$$

$$\}$$

У каждой оси запроса есть номер: 0 для оси X, 1 для оси Y и т. д. В указанном раннее синтаксисе инструкции значение Integer Expression задает номер оси. В запросе многомерных выражений можно указать до 128 осей. Наиболее часто в запросах встречается до пяти осей, поэтому для них созданы специальные псевдонимы COLUMNS, ROWS, PAGES, SECTIONS, CHAPTERS.

Каждый запрос многомерных выражений выполняется для конкретно заданного куба. Контекст куба определяет элементы, которые вычисляются в выражениях запроса.

В инструкции SELECT контекст куба задается с помощью предложения FROM, где в качестве контекста может выступать не только куб целиком, но и его вложенный куб. Указание контекста позволяет расширять или ограничивать его с использованием дополнительных функций.

Предложение FROM определяется следующим синтаксисом:

$$\langle SELECT \text{ subcube clause} \rangle ::=$$

$$\text{ Cube Identifier } |$$

$$(SELECT [*]$$

$$[\langle SELECT \text{ query axis clause} \rangle [, \langle SELECT \text{ query axis clause} \rangle \dots]]$$

$$FROM \langle SELECT \text{ subcube clause} \rangle \langle SELECT \text{ slicer axis clause} \rangle)$$

В указанном ранее синтаксисе куб (вложенный куб), над которым производится операция выборки, описывается предложением *<SELECT subcube clause>*.

После выполнения инструкции SELECT ось среза производит фильтрацию данных. При этом в конечном итоге получают данные, имеющие пересечения с заданными элементами. Ось среза определяется в предложении WHERE инструкции SELECT.

Для явного определения оси среза используется следующий синтаксис:

(SELECT slicer axis clause) ::= WHERE Set Expression

В приведенном синтаксисе определения оси среза аргумент SET EXPRESSION может принимать либо кортежное выражение, интерпретируемое как набор при вычислении предложения, либо выражение набора. Если задано выражение набора, язык многомерных выражений пытается вычислить набор с использованием статистического вычисления по результирующим ячейкам в каждом кортеже набора. Иными словами, используется функция Aggregate для набора, при этом каждая мера статистически обрабатывается с помощью связанной статистической функции. Кроме того, если выражение набора нельзя выразить в виде перекрестного соединения элементов иерархии атрибутов, при вычислении в языке многомерных выражений ячейки, находящиеся за пределами выражения набора, с помощью которого определена ось среза, интерпретируются, как имеющие значения NULL.

Таким образом, упрощенный запрос на выборку данных из многомерного куба включает в себя следующие ключевые слова: SELECT, FROM, ON COLUMNS, ON ROWS. При этом в предложении FROM указывается имя куба, а каждый показатель оси представляется в виде *[Dimension].[Member]*. Если в выборке участвует несколько членов, то они должны быть сгруппированы и записаны согласно следующему синтаксису:

{[Dimension].[Member],[Dimension].[Member][...]}

Реализуемый запрос на выборку данных из многомерных OLAP-кубов с помощью графического интерфейса реализуется в два этапа: получение данных и получение метаданных из системы управления данными куба. Реализация этапов приведена в алгоритмах получения метаданных и данных куба.

Алгоритмы интерактивного формирования MDX-запроса к данным OLAP-куба

Алгоритм получения метаданных куба для построения запроса на выборку. При динамической генерации MDX-запроса на выборку необходимо обеспечить корректную последовательность ввода элементов запроса. Прежде всего необходимо указать базу данных, с которой будет происходить взаимодействие. После чего можно приступить к формированию запроса.

Для того чтобы составить запрос, необходимо получить соответствующие метаданные. В контексте многомерных баз данных и, в частности, кубов, метаданными будут являться следующие компоненты: список кубов выбранной базы данных; список измерений, характеризующих куб; список членов, содержащихся в измерении. Учитывая, что минимально допустимый запрос на выборку имеет вид SELECT FROM Cube (где Cube – имя куба в выбранной базе данных), то для такого запроса необходимо получить только список кубов. Для развернутого запроса требуется получить и ограничить список мер, доступных в каждом кубе.

Общая схема получения метаданных для построения запроса на выборку состоит, с учетом особенностей реализации клиент-серверной технологии обращения к OLAP-серверу, в реализации следующих шагов.

1. Запросить список кубов, находящихся в выбранной базе данных.
2. Выбрать куб из списка, сформированного в п. 1, для заполнения предложения FROM.
3. Запросить список размерностей для куба из п. 2.
4. Выбрать размерность из списка, сформированного в п. 3.

5. Запросить меры для выбранной из п. 4 размерности.

6. Выбрать одну из мер, доступных в п. 5, сформировав тем самым одну из инструкций.

7. Повторить п. 3–6 требуемое количество раз для формирования инструкций вида *<SELECT query axis clause>* и, соответственно, определения осей результирующего запроса.

8. Повторить п. 3–6 требуемое количество раз для формирования инструкций вида *<SELECT slicer axis clause>* и, соответственно, определения осей среза.

Блок-схема алгоритма ввода метаданных представлена на рис. 1. Результатом работы алгоритма является сформированный MDX-запрос идентификаторов кубов, размерностей и мер для выбранной системы управления базами данных.

Алгоритм получения данных MDX-запроса. После формирования запроса (были указаны значения куба, размерностей и соответствующих осей), требуется проверить его на синтаксическую корректность, выполнить и продемонстрировать результаты. Эти этапы реализуются в алгоритме получения данных по ранее построенному запросу.

На первом этапе осуществляется проверка синтаксиса полученного запроса. Для выполнения простейшего запроса на выборку необходимо и достаточно присутствие трех лексем: ключевые слова SELECT FROM и имя куба. Следует отметить, что в конечном запросе должны участвовать все размерности куба. Поэтому, расширяя минимально допустимый запрос путем добавления описания осей, можно столкнуться с проблемой, когда какая-либо размерность не была указана. В этом случае существуют два способа устранения проблемы. Первый из них заключается в выборе одной меры по умолчанию для каждой неуказанной размерности. Второй использует в запросе все меры неуказанной размерности, т. е. предполагается, что пользователь заведомо хочет выбрать все данные из этой размерности. Выбор варианта реализации отбора мет определяется выбором пользователя для каждого запроса.

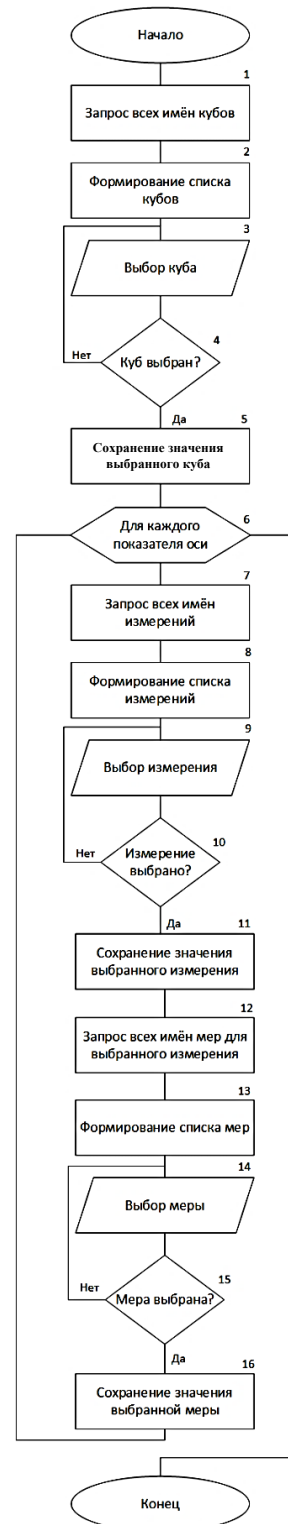


Рис. 1. Алгоритм получения метаданных для MDX-запроса

Fig. 1. The Algorithm of metadata creating for MDX-query

Общая схема получения данных в форме MDX-запроса на выборку состоит, с учетом особенностей реализации графического интерфейса представления OLAP-запроса и клиент-серверной технологии обращения к OLAP-серверу, из следующих шагов.

1. Запрос имени базы данных. Найти структуру в интерфейсе, отвечающую за выбор базы данных, и получить текущее значение имени БД.

2. Запрос имени куба. Найти структуру в интерфейсе, отвечающую за выбор куба (предложение FROM), и получить текущее значение имени куба.

3. Запрос идентификатора сессии. Отправить web-запрос подключения к удаленному серверу и забрать sid (идентификатор сессии).

4. Формирование кортежей, определяющих элементы куба.

5. Запрос результата выполнения MDX-запроса. Сформировать web-запрос на основе полученных данных и отправить его на выполнение к удаленному серверу.

6. Формирование результирующего набора данных. Синтаксический разбор полученного результата выполнения web-запроса для извлечения данных.

7. Формирование итогового MDX-запроса. Получить все пары <Измерение, Мера>, использованные при построении запроса, и составить синтаксически правильный MDX-запрос.

8. Вывод текста запроса и полученных данных. Вывод результирующего набора

данных и полученного MDX-запроса на экран.

Алгоритм получения данных из графического приложения MDX-запроса представлен на рис. 2 Псевдокод формирования текста запроса приведен в листинге.

Результатом работы алгоритма является текст MDX-запроса и данные, полученные в результате его исполнения.

Реализация подсистемы генерации MDX-запроса. Для реализации подсистемы генерации MDX-запроса разработан прототип программного обеспечения. Цель его функционирования состоит в предоставлении интерактивного графического интерфейса для генерации MDX-запроса на выборку данных из OLAP-сервера Palo, с последующим выводом результатов и возможностью просмотра сгенерированного текста запроса. Общая схема функционирования среды представлена на рис. 3. Для выполнения запроса выполняются следующие шаги.

1. Клиентское программное обеспечение эмулирует режим работы MDX, который реализует функциональную возможность выборки данных из многомерных OLAP-кубов.

2. С помощью web-запросов к MOLAP-серверу клиент получает метаданные из многомерных баз данных и формирует структуру запроса на выборку.

3. Конечный MDX-запрос преобразуется в web-запрос и посредством WEB API извлекает требуемые данные из выбранного многомерного OLAP-куба.

1 ПОЛУЧИТЬ список пар <Размерность, Мера> *PairList*;

2 СГРУППИРОВАТЬ меры из *PairList* по размерностям в *MeasureList*

3 *for (i = 0; i < MeasureList.size (); i++)*

4 ЕСЛИ (*MeasureList [i] == null*) ТО

5 ДОБАВИТЬ в *MeasureList* все идентификаторы i-й размерности;

6 СОСТАВИТЬ комбинации мер в установленном порядке для формирования кортежей, определяющих положение элемента в кубе.

Процедура формирования MDX-запроса
The MDX-query forming procedure

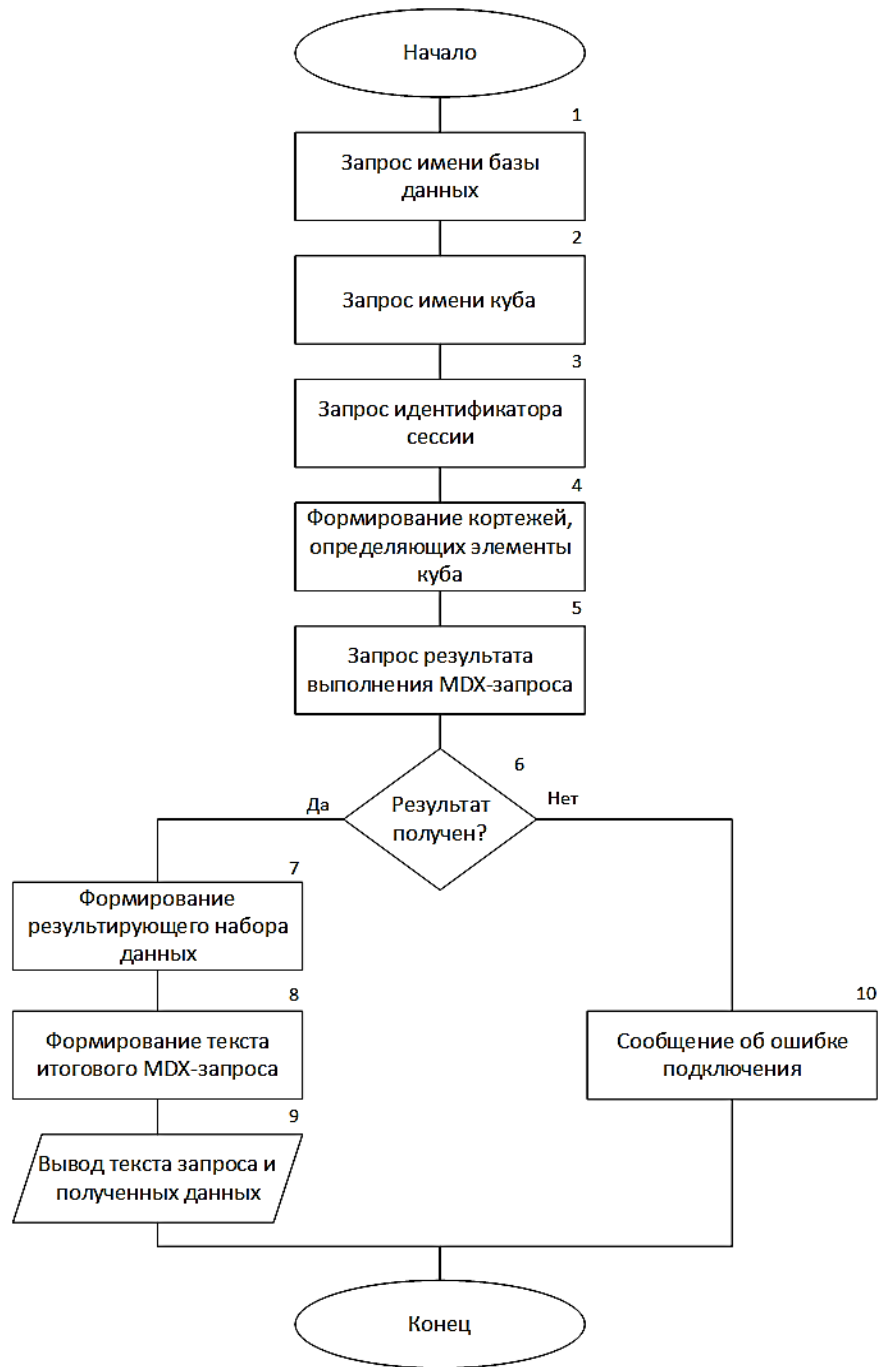


Рис. 2. Алгоритм получения текста и результата MDX-запроса
Fig. 2. The algorithm of text and results MDX-query receiving

Прежде всего для формирования запроса необходимо узнать список баз данных, доступных для работы на сервере. Подключение к MOLAP-серверу осуществляется автоматически при выполнении каждого запроса. Такой подход обеспечивает наличие актуального номера сессии (sid),

чтобы не приходилось его периодически обновлять по истечении тайм-аута. Таким образом, процедура формирования запроса начинается с выбора базы данных из выпадающего списка. После чего пользователю становится доступно основное окно запроса.

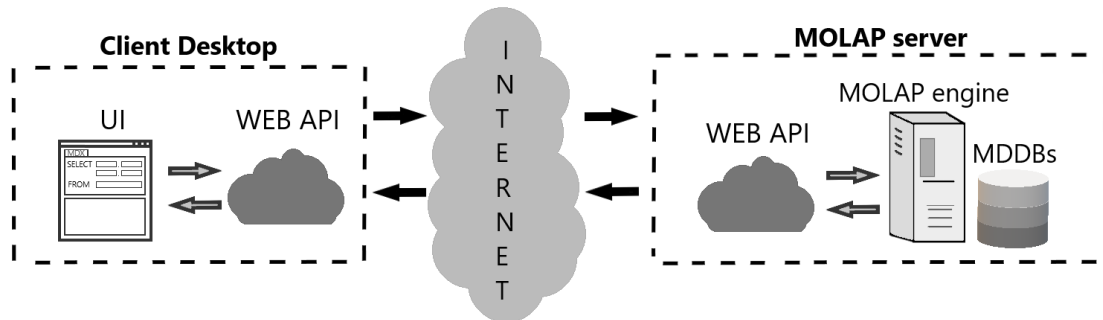


Рис. 3. Общая схема взаимодействия приложения с сервером
Fig. 3. The server and client interconnection general scheme

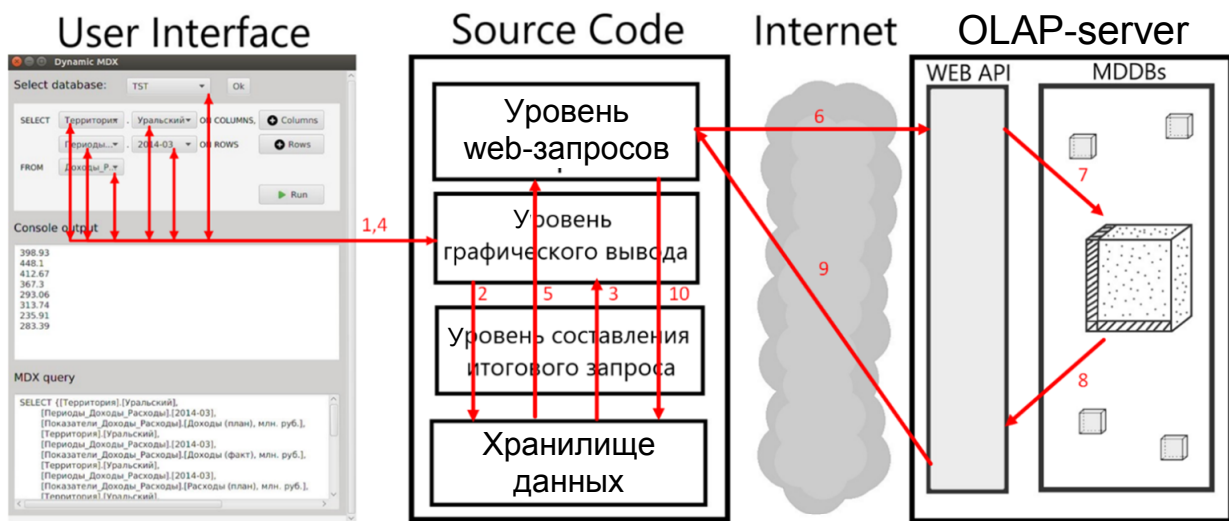


Рис. 4. Схема получения и представления метаданных запроса
Fig. 4. The query metadata getting and representation scheme

На рис. 4 представлена схема получения метаданных для построения запроса. Схема содержит следующие шаги.

1. Выбрать элемент запроса, для которого необходимо получить метаданные.
2. Проверить наличие запрашиваемых метаданных в хранилище.
3. Направить полученный список значений на уровень графического вывода.
4. Поместить полученный список значений в соответствующую структуру из п. 1.
5. Сформировать текст web-запроса.
6. Отправить сформированный запрос к OLAP-серверу.
7. Запросить метаданные из многомерного куба.
8. Передать данные в WEB API.
9. Получить ответ от OLAP-сервера.
10. Произвести синтаксический разбор ответа и сохранить значения в хранилище.

Получение метаданных может быть осуществлено двумя способами.

1. Если в локальном хранилище есть запрашиваемые данные, то маршрут движения по схеме будет определяться следующими шагами: 1, 2, 3, 4.

2. Если в локальном хранилище не оказалось запрашиваемых данных, то необходимо выполнять соответствующий web-запрос. В этом случае маршрут будет выглядеть так: 1, 2, 5, 6, 7, 8, 9, 10, 3, 4.

На рис. 5 представлена схема получения данных для сформированного MDX-запроса. Схема содержит следующие шаги.

1. Забрать текущие значения показателей осей из графического интерфейса.
2. Сохранить полученные значения в хранилище данных.
3. Направить полученные значения на уровень составления итогового запроса.

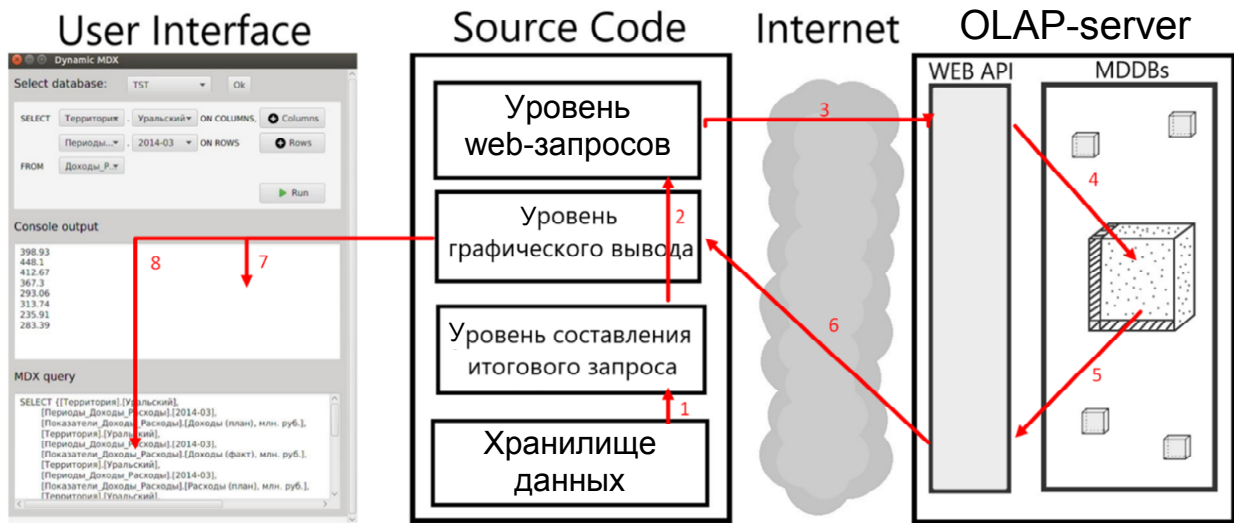


Рис. 5. Схема получения и представления данных запроса
Fig. 5. The data getting and representation scheme

4. Проверить запрос и составить кортежи, определяющие элементы куба.
5. Отправить web-запрос к OLAP-серверу.
6. Запросить данные из многомерного куба.
7. Передать данные в WEB API.
8. Получить ответ от OLAP-сервера.
9. Направить полученные результаты на уровень графического вывода.
10. Вывести на экран полученные значения ячеек.
11. Вывести на экран текст MDX-запроса.

На рис. 6 показан пример интерфейса, заполненный исходными метаданными текстом сформированного запроса и данными, полученными в результате его исполнения.

Согласно алгоритму получения метаданных для построения запроса на выборку, сначала необходимо получить список кубов и выбрать один из предложенных вариантов. Как только пользователь выбирает куб, в поля измерений автоматически подгружаются соответствующие метаданные.

Поле выбора меры измерения остается неактивным до тех пор, пока не будет выбрано соответствующее значение измерения. При этом, как только пользователь выбирает размерность, происходит формирование и отправка web-запроса к MOLAP-

серверу, который должен забрать из базы значения мер. Следует отметить, что значения мер не хранятся в локальном репозитории, а автоматически подгружаются из многомерной БД. Эта операция длится в среднем 40 мс и позволяет не хранить большие объемы метаданных локально.

После заполнения всех полей необходимо составить кортежи, состоящие из идентификаторов мер, для формирования конечного web-запроса. После его выполнения на экран пользователя выводятся результаты и сгенерированный текст MDX-запроса.

Для представления данных используются редактируемые и не редактируемые области. Редактируемые текстовые области: область 2 содержит выбор доступных кубов; область 3 – выбор показателей и мер; 5 – результат выполнения запроса. Области 1, 4 и 6 являются информационными и не подлежат редактированию.

Часть метаданных, полученная с удаленного сервера, хранится локально. К ней относятся имена БД, кубов, а также измерений. Такое решение позволяет сократить количество обращений с помощью web-запросов, тем самым уменьшая время отклика программы. Поскольку не все меры используются при построении запроса, то для каждого выбранного измерения формируется соответствующий запрос для получения списка имен мер.

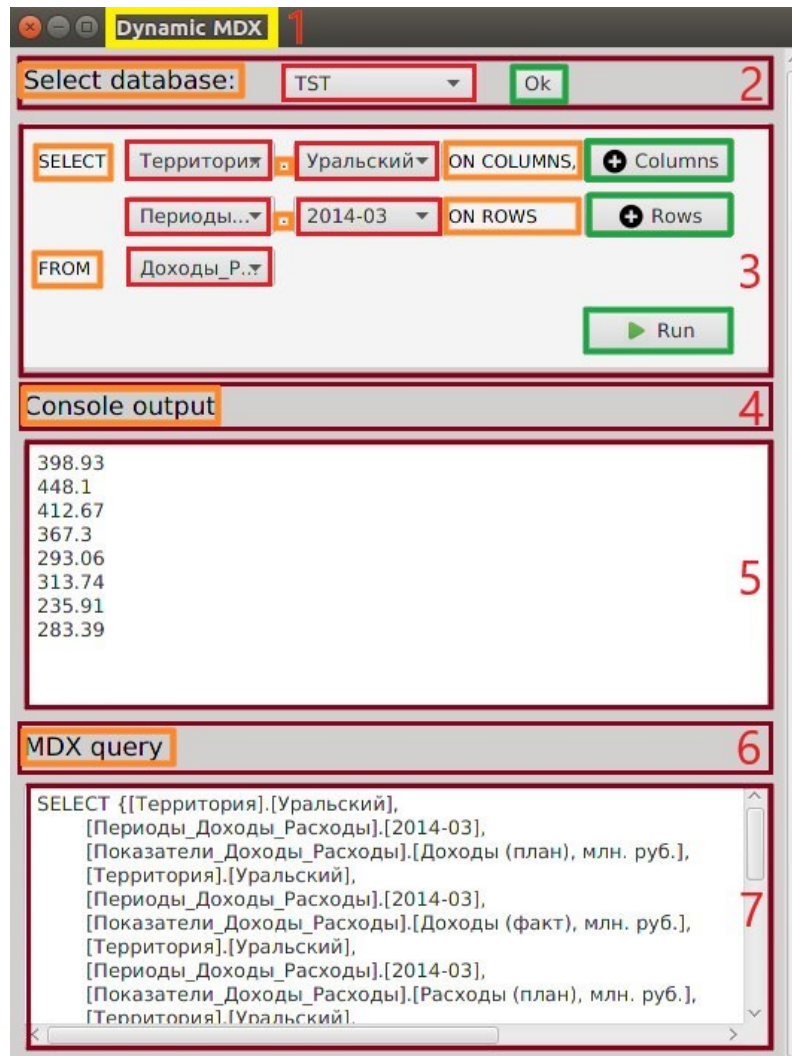


Рис. 6. Структурное содержание программного интерфейса
Fig. 6. The structure of user interface

Для хранения получаемых метаданных используются списки строкового типа. Что касается данных, поступающих после выполнения итогового MDX-запроса, то они направляются сразу в текстовое поле интерфейса без промежуточного хранения, откуда при необходимости могут быть извлечены.

Предложенный интерфейс обеспечивает графическое представление и формирование запроса на выборку данных из многомерного куба, результаты которого могут применяться для интерактивного отображения в приложениях пользователя.

Исследование времени выполнения MDX-запроса

Для реализованного прототипа системы интерактивного запроса данных требуется получить и проанализировать временные зависимости выполнения MDX-запроса, что позволит оценить скорость формирования запроса и получения данных с удаленного OLAP-сервера.

Постановка эксперимента. Для проведения экспериментов была выбрана тестовая база данных и куб данных «Доходы Расходы», структура которого представлена на рис. 7.

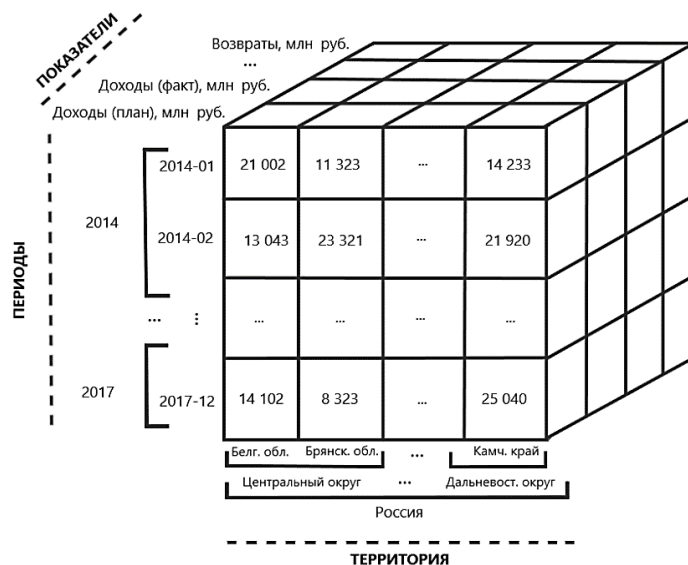


Рис. 7. Структура трехмерного куба данных «Доходы Расходы»
 Fig. 7. The structure of thredementional «Income and expenses» cube

В ходе эксперимента необходимо определить время подготовки запроса и время выполнения запроса данных, а также общее время работы алгоритма, что позволит оценить объем данных, доступных для интерактивного запроса. Параметры и константы эксперимента приведены в таблице.

Параметры и константы эксперимента
The experiment constants and parameters

Наименование параметра	Значение
Сервер	
OLAP-сервер	Palo
Способ хранения данных	MOLAP
Тип интерфейса	WEB API
Куб	
Название	Доходы Расходы
Количество измерений	3
Общее количество мер	155
Общее число ячеек	32640
Количество заполненных ячеек	30515
Максимальное количество уровней иерархии	3

Выбранный куб содержит три измерения: «Территория», «Периоды», «Показатели», два из которых представляют собой

иерархию мер с максимальной глубиной, равной трем.

Для представленного куба формировался запрос, содержащий разное число мер и возвращающий различное число ячеек.

Обработка и анализ результатов. Для анализа времени выполнения программного кода были выделены участки программного кода: подготовки метаданных для составления web-запроса, выполнения запроса с учетом времени передачи и приема данных.

На рис. 8 представлена зависимость времени выполнения блоков кода от количества полученных данных и числа мер.

Одной из характеристик интерактивного графического интерфейса является время отклика. Для профессионального программного продукта крайне важно, чтобы большая часть операций выполнялась за минимально допустимое время. Анализируя график, изображенный на рисунке, можно сделать вывод о том, что краткосрочные запросы, которые выполняются менее 0,5 с, будут возвращать до 1650 значений. Запросы средней продолжительности, время выполнения которых варьируется от 0,5 до 2 с, возвращают до 6000 значений. Таким образом, для того чтобы выполнять запросы на выборку данных за приемлемое время, желательно извлекать не более 6000 значений из многомерного OLAP-куба.

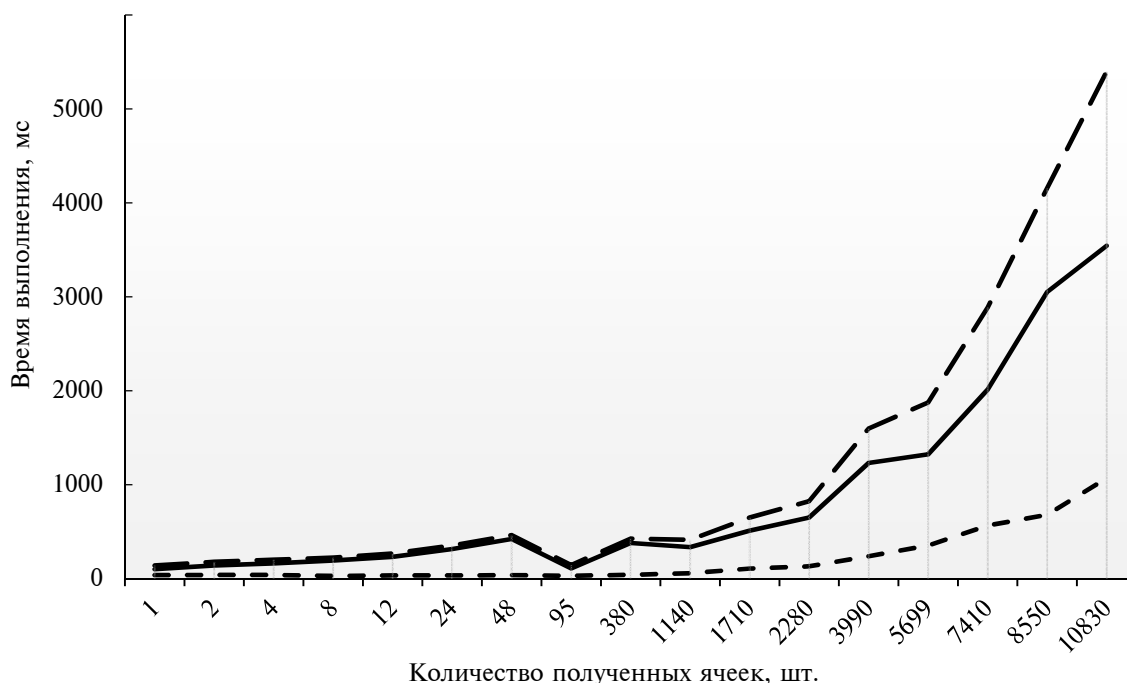


Рис. 8. Зависимость времени выполнения блока кода от количества полученных данных (—) подготовка запроса; (---) выполнение запроса; (- - -) общее время

Fig. 8. The dependence between code running time and data volume

Заключение

Рассмотрены технология генерации MDX-запроса на выборку данных из OLAP-кубов сервера Palo с интерактивным графическим интерфейсом, средства динамической генерации MDX-запросов, которые позволили выявить основные компоненты, необходимые для реализации графического приложения.

С помощью проведенного синтаксического анализа MDX-запроса на выборку данных разработаны алгоритмы получения метаданных и данных из многомерного куба на основе сгенерированного MDX-запроса.

Разработанные алгоритмы реализованы с использованием интерактивного графического интерфейса, позволяющего динамически генерировать текст MDX-запроса, выполнять его на удаленном OLAP-сервере, выводить полученные данные в специальное текстовое поле, а также просматривать текст результирующего MDX-запроса на выборку данных.

Реализован прототип подсистемы генерации MDX-запроса на выборку данных из OLAP-кубов сервера Palo с интерактивным графическим интерфейсом.

Для тестирования построенной подсистемы динамической генерации MDX-запросов выбраны условия проведения эксперимента, целью которого являлось определение объема получаемых данных, обеспечивающего приемлемую скорость выполнения запроса.

Реализованные алгоритмы получения данных и метаданных из многомерных кубов удаленного сервера продемонстрировали работоспособность прототипа в целом. Дальнейшим направлением исследования может стать расширение функциональных возможностей разработанного приложения.

Работа подготовлена в ходе реализации комплексного проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0259 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. **Чибисов В.Н., Суслов И.А., Маркилов И.А.** Построение OLAP-кубов с помощью стандартных средств разработки web-приложений // Программные продукты и системы. 2013. № 4.
2. **Журавлев А.Е.** Механизм семантического анализатора запросов к базам данных информационных систем // Вестник Санкт-Петербургского университета. Серия 10. Прикладная математика. Информатика. Процессы управления. 2012. № 4.
3. **Patil A.D., Gangadhar N.D.** OLaaS: OLAP as a Service // Cloud Computing in Emerging Markets, 2016 IEEE Internat. Conf. on. IEEE, 2016. Pp. 119–124.
4. **Шмелева Д.В. и др.** Реляционная OLAP-система архивных метеоданных на базе СПО (свободного программного обеспечения) // Научный альманах. 2017. № 5-3. С. 143–146.
5. **Li J., et al.** The design and implementation of web-based OLAP drilling analysis system // Fuzzy Systems and Knowledge Discovery, 2010 7th Internat. Conf. on. IEEE, 2010. Vol. 6. Pp. 2570–2573.
6. **Vaisman A.A., Izquierdo A., Ktenas M.** Web-enabled temporal OLAP // Web Congress, 2006. LA-Web'06. 4th Latin American. IEEE, 2006. Pp. 220–229.
7. **Zhang P., Xi J.** Attribute-driven design of MDX compiler // Computer Science and Software Engineering. 2008 Internat. Conf. on. IEEE, 2008. Vol. 2. Pp. 508–511.
8. **Лобел Л.** Разработка приложений на основе Microsoft SQL Server 2008. М.: Русская редакция, 2010.
9. **Петров С.А.** Метод реализации статистических отчетов с поддержкой детализации // Cloud of science. 2018. Vol. 5. No. 4. Pp. 765–777.
10. **Чибисов В.Н., Суслов И.А., Маркилов И.А.** Построение OLAP-кубов с помощью стандартных средств разработки web-приложений // Программные продукты и системы. 2013. № 4.
11. **Наумов В.Н., Лычагина Е.Б., Шарабабаева Л.Ю.** Использование BI-систем для обеспечения информационно-аналитической деятельности органов государственной власти // Управленческое консультирование. 2016. № 3(87). С. 144–153.
12. **Bhowmik T., Sarkar A., Debnath N.C.** OLAP umbrella: Visualization model for multidimensional databases // ACS/IEEE Internat. Conf. on Computer Systems and Applications. Hammamet, 2010. Pp. 1–8.
13. **Трофимов С.П., Пономарева О.А.** Формализация OLAP-кубов и реляционных связей в базах данных с помощью алгебры множеств // Компьютерный анализ изображений: Интеллектуальные решения в промышленных сетях – 2016. Екатеринбург, 2016. С. 230–233.
14. **Артюшина Е.А., Сальников И.И.** Технологии in-темогу для хранения, обработки и анализа больших объемов, структурированных и слабоструктурированных данных // XXI век: итоги прошлого и проблемы настоящего плюс. 2018. Т. 7. № 4. С. 147–152.
15. **Ноженкова Л.Ф., Шайдуров В.В.** OLAP-технологии оперативной информационно-аналитической поддержки организационного управления // Информационные технологии и вычислительные системы. 2010. № 2. С. 15–27.

Статья поступила в редакцию 12.11.2018.

REFERENCES

1. **Chibisov V.N., Suslov I.A., Markilov I.A.** Building OLAP-cubes using standard web-development applications. *Programmnyye produkty i sistemy [Software&Systems]*, 2013, No. 4. (rus)
2. **Zhuravlev A.Ye.** Queries semantic analyzer to information systems databases technol. *Vestnik of St. Petersburg State University. Series 10. Applied Mathematics. Computer Science. Control Processes*, 2012, No. 4. (rus)
3. **Patil A.D., Gangadhar N.D.** OLaaS: OLAP as a Service. *Cloud Computing in Emerging Markets, 2016 IEEE International Conference on*, IEEE, 2016, Pp. 119–124.
4. **Shmeleva D.V., et al.** Relational OLAP system is the archival of meteorological data on the basis of the SPO (the free software). *Science Almanac*, 2017, No. 5-3, Pp. 143–146. (rus)
5. **Li J., et al.** The design and implementation of web-based OLAP drilling analysis system. *Fuzzy Systems and Knowledge Discovery, 2010 7th International Conference on*, IEEE, 2010, Vol. 6, Pp. 2570–2573.
6. **Vaisman A.A., Izquierdo A., Ktenas M.** Web-enabled temporal OLAP. *Web Congress, 2006. LA-Web'06. 4th Latin American*. IEEE, 2006, Pp. 220–229.
7. **Zhang P., Xi J.** Attribute-driven design of MDX compiler. *Computer Science and Software Engineering, 2008 International Conference on*. IEEE, 2008, Vol. 2, Pp. 508–511.

8. **Lobel L.** *Razrabotka prilozheniy na osnove Microsoft SQL Server 2008 [Application Development Based on Microsoft SQL Server 2008]*. Moscow: Russkaya redaktsiya Publ., 2010. (rus)
9. **Petrov S.A.** Metod realizatsii statisticheskikh otchetov s podderzhkoy detalizatsii. *Cloud of science*, 2018, Vol. 5, No. 4, Pp. 765–777. (rus)
10. **Chibisov V.N., Suslov I.A., Markilov I.A.** The method of implementation of statistical reports with support for detail. *Programmnyye produkty i sistemy*, 2013, No. 4. (rus)
11. **Naumov V.N., Lychagina Ye.B., Sharababayeva L.Yu.** Use of BI-Systems for ensuring information and analytical activity of public authorities. *Administrative Consulting*, 2016, No. 3(87), Pp. 144–153. (rus)
12. **Bhowmik T., Sarkar A., Debnath N.C.** OLAP umbrella: Visualization model for multidimensional databases. *ACS/IEEE International Conference on Computer Systems and Applications – 2010*, Hammamet, 2010, Pp. 1–8.
13. **Trofimov S.P., Ponomareva O.A.** Formalization of OLAP-cubes and relational databases with the algebra of sets. *Computer image analysis: Intelligent solutions in industrial networks – 2016*, Yekaterinburg, 2016, Pp. 230–233. (rus)
14. **Artyushina Ye.A., Salnikov I.I.** In-memory technologies for storing, processing and analysing of large volumes of structured and weakly structured data. *XXI Century: Resumes of the Past and Challenges of the Present plus*, 2018, Vol. 7, No. 4, Pp. 147–152. (rus)
15. **Nozhenkova L.F., Shaydurov V.V.** OLAP-tehnologii operativnoy informatsionno-analiticheskoy podderzhki organizatsionnogo upravleniya [OLAP-technology operational information and analytical support for organizational management]. *Computer Science and Control*, 2010, No. 2, Pp. 15–27. (rus)

Received 12.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПОПОВ Сергей Геннадьевич

POPOV Sergey G.

E-mail: popovserge@spbstu.ru

ЛИСЕНКОВА Анна Алексеевна

LISENKOVA Anna A.

E-mail: nutochka97@gmail.com