

ОПЕРАТИВНАЯ АНАЛИТИЧЕСКАЯ ОБРАБОТКА ДАННЫХ ОБ АКТИВНОСТИ СТУДЕНТА В СРЕДЕ КОМПЬЮТЕРНОГО ОБУЧЕНИЯ

ПАНТЕЛЕЕВ Е.Р., д-р техн. наук, СУВОРОВ В.А., студ.

Рассматриваются вопросы разработки комплекса методов сбора и оперативной аналитической обработки данных об активности студента в среде компьютерного обучения. Представлены архитектура комплекса, модель хранения данных, программный интерфейс заполнения хранилища. Описаны методы и Web-интерфейс аналитической обработки. Приведены примеры применения комплекса для анализа результатов компьютерного обучения.

Ключевые слова: компьютерное обучение, активность студента, оценка знаний.

OPERATIVE ANALYTIC DATA PROCESSING ON STUDENT'S ACTIVITY IN COMPTUPER-AIDED TEACHING ENVIRONMENT

PANTELEEV E.R., Ph.D., SUVOROV V.A., student.

The article concerns the issues of complex development devoted to data collection and operative analytic processing of data on student's activity in computer-aided teaching environment. It gives the complex architecture, data storage model, program interface of archive filling. The methods and web-interface of analytic processing are described. It gives the examples of complex application for the analysis of computer-aided teaching results.

Key words: computer-aided teaching, student's activity, knowledge assessment.

Отсутствие непосредственного контакта студента и преподавателя в компьютерных системах обучения делает особенно актуальной задачу адаптации информационных ресурсов обучения в соответствии с целями и персональным уровнем подготовки. Построение адаптационных механизмов на базе оперативной транзакционной обработки персональных действий ограничивает возможности управления обучением уровнем персональной адаптации материала учебных программ (управление навигацией, визуальным представлением материала, схемами контроля знаний). Транзакционная обработка не позволяет выявить эффективность использования информационных ресурсов в контексте целей компьютерного обучения. Это существенно затрудняет решение задач компоновки информационных ресурсов обучения и контроля, обеспечивающих возможности для достижения эталонного уровня подготовки во внешнем контуре схемы «планирование – реализация плана качества» [1].

Адекватную информационную основу для решения этих задач дает использование хранилищ данных и технологий их оперативной аналитической обработки (online analytical processing, OLAP) [2]. Примерами немногочисленных пока попыток использования OLAP-технологий для анализа эффективности являются работы [3, 4]. Недостатком предложенного в [4] решения является его интеграция в конкретную интеллектуальную обучающую систему (ИОС) и невозможность повторного использования в других системах. Преимущество повторного использования достигается применением клиент-серверных архитектур хранения и обработки данных об активности пользователя. Их характерные особенности [5]:

- информация о действиях пользователя размещена в интегрированном хранилище и может быть одновременно предоставлена нескольким приложениям;

- информация, размещенная в хранилище одним приложением, может быть предоставлена другому приложению;

- информация о действиях пользователя хранится в избыточном и непротиворечивом формате;

- сервер предоставляет пользователю средства защиты данных и контроля доступа.

Один из вариантов реализации клиент-серверной архитектуры хранилища данных в сети Интернет (User Modeling Server CUMULATE) [6] предоставляет пользователю программный Web-интерфейс наполнения хранилища. Однако интерфейсы аналитической обработки в рамках этого проекта не реализованы. Между тем потребность в разработке методов и средств Web-OLAP, ориентированных на решение задач анализа процессов и результатов Web-обучения в их взаимосвязи, безусловно, существует. Практический интерес представляют ответы, например, на такие вопросы:

- Как распределены дефекты подготовки на оверлейной модели студента (группы студентов)?

- Какова дискриминативность и надежность процедур контроля?

- Как соотносятся результаты подготовки студентов по различным программам (или результаты по одной и той же программе по годам и/или группам)?

- Какова эффективность процедур тренажа, выраженная отношением процентов правильных ответов (с возможной фильтрацией по концептам онтологической модели) в режимах тренажа и контроля?

- Какова эффективность управления навигацией, выраженная отношением количества активированных ссылок к количеству правильных ответов?

С учетом содержательного аспекта Web-OLAP, можно считать актуальной задачу разработки хранилища и средств OLAP, которые удовлетворяют следующим требованиям [8]:

- наличие программного интерфейса заполнения хранилища;

- наличие пользовательского интерфейса для построения нерегламентированных запросов к хранилищу данных;

- оперативное выполнение запросов;
- открытые спецификации на интерфейсы, сервисы и поддерживаемые форматы данных;
- минимальный трафик «клиент-сервер»;
- нулевое администрирование клиента OLAP, простота освоения инструментов анализа;
- наглядное представление результатов выполнения запросов средствами деловой графики.

Вопросам разработки клиент-серверной архитектуры хранилища, программных интерфейсов его заполнения, а также Web-интерфейсов аналитической обработки, ориентированных на пользователя с отсутствием навыков работы с OLAP-приложениями и разнообразными потребностями в анализе активности студентов в среде Web-обучения (авторы курсов, кураторы программ обучения, администраторы), и посвящена настоящая статья.

Взаимодействие компонентов комплекса оперативной аналитической обработки со средой Web-обучения (рис. 1), собственно говоря, имеет место только на этапе сбора информации, когда информация о действиях студента, представленная в формате событий, поступает на OLAP-сервер (сервер аналитики), который размещает ее в хранилище OLAP. На этапе аналитической обработки данных подсистема обучения не связана с подсистемой аналитики ни программно, ни через разделяемую информацию. Данные о событиях могут быть извлечены из хранилища и представлены в удобной для анализа форме при помощи Web-интерфейса клиента аналитики. Отсутствие взаимодействия системы Web-обучения с сервером аналитики на этапе обработки данных оправдано самим характером решений, принимаемых на основе OLAP. Это, как правило, решения контура планирования стандартов PDCA [0], связанные с пересмотром приоритетов целей обучения и соответствующим перераспределением информационных ресурсов, которые принимаются в режиме «офлайн».

Рассмотрим модели и методы решения задач сбора и аналитической обработки данных, реализованные в рамках предлагаемого подхода, более подробно.

Информационную основу решения задачи сбора и обработки данных о действиях студента в среде Web-обучения составляет модель его активности, представленная в виде упорядоченной во времени последовательности дискретных событий.

Под *событием* E_i понимается акт взаимодействия студента с интерактивным элементом учебного или контролирующего материала учебника, который инициирует передачу параметров этого взаимодействия на сервер. К таким взаимодействиям относятся активация навигационной ссылки на учебный материал или задание контроля, а также нажатие кнопки на форме контроля знаний. Поэтому целесообразно различать следующие типы событий (в скобках указаны их атрибуты A_i и измерения D_i):

- активизация ссылки (время, пользователь, группа, программа, тип источника ссылки, объект-адресат ссылки);
- ответ на вопрос (время, вопрос, пользователь, группа, программа, модуль, оцениваемый элемент знаний, мера правильности ответа);
- инициализация/завершение процедуры контроля (время, флаг своевременности, тип события, пользователь, группа, программа, модуль).

Поддерживаемые сервером аналитики типы дискретных событий и их измерения представлены на рис. 2 (кроме явных атрибутов и измерений сервер аналитики в качестве «скрытого» измерения фиксирует IP-адрес клиента, что позволяет при аналитической обработке фильтровать события по их источнику).

Программный интерфейс сбора статистики представлен на сервере аналитики сценарием обработки событий. Передача этому сценарию данных, предварительно преобразованных к формату URL, осуществляется по протоколу HTTP (методы POST или GET).

При отсутствии хотя бы одного необходимого параметра или при недопустимом значении любого переданного параметра запись события в базу не производится.

Как следует из приведенных выше определенных типов событий, каждое событие E_i определено в своем базисе измерений D_i : $E_i = f(D_i)$, причем $\forall i, j: D_i \cap D_j \neq \emptyset$. Это обстоятельство определило выбор поликубической модели хранилища в виде нескольких гиперкубов с различной размерностью и с различными измерениями в качестве их граней (рис. 2). Такое решение более экономично по сравнению с «монолитным» гиперкубом, так как позволяет сократить количество «пустых» ячеек.

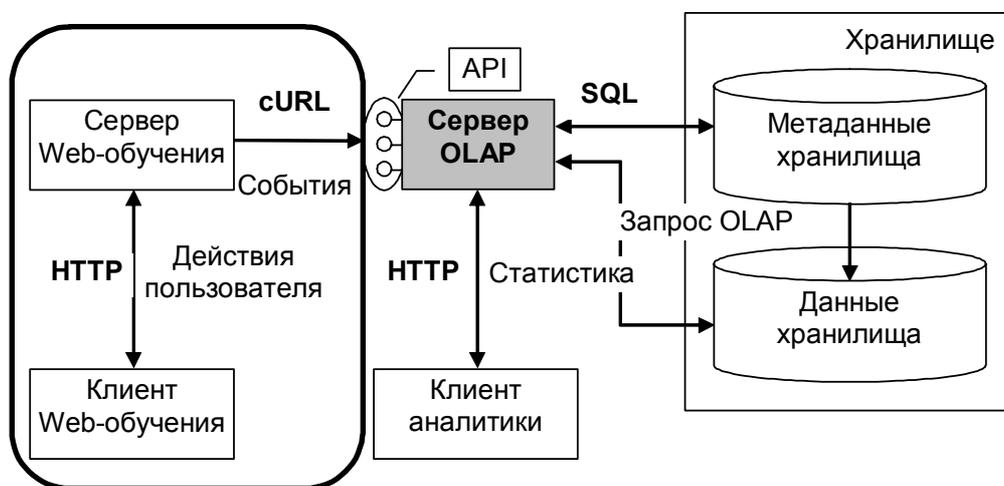


Рис. 1. Взаимодействие средств OLAP со средой Web-обучения

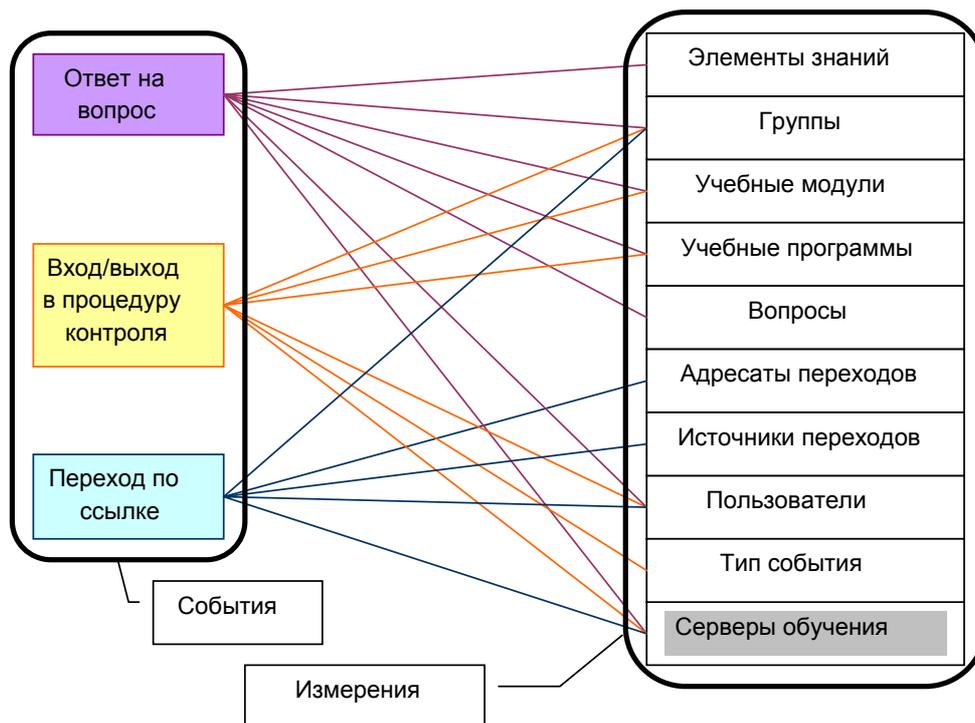


Рис. 2. Виды активности студента и их атрибуты

Web-интерфейс аналитической обработки (рис. 3, средняя часть) включает в себя следующие основные элементы:

- панель управления для выбора типа события, относительно которого выполняется запрос, и настройки параметров построителя запросов (размеры диаграммы, глубина трехмерной прорисовки, цвет заголовка, цвета фона картинки, цвета фона диаграммы, цвета линий сетки, цвет бордюра, шрифт заголовка, количество рядов подписей и угол их наклона);
- панель запросов для выбранного типа события, включающую три закладки:
 - Мастер запросов;
 - SQL-консоль;
 - Сохраненные выборки.

Наибольший интерес представляет закладка «Мастер запросов». Запрос пользователя формируется путем заполнения стандартного SQL-шаблона `SELECT...FROM...WHERE...GROUP BY...` информацией об анализируемых событиях, их измерениях и способах агрегирования атрибутов. Информационную базу решения этой задачи составляет метаописание взаимосвязи событий E и измерений D поликуба. Метаописание реализовано в виде явно определенного отношения $R : (E \times D) \rightarrow \{', утверждение_SQL\}$, областью значений которого являются фрагменты SQL-утверждений, используемые для заполнения шаблона. Реляционная модель метаописания представлена в верхней части рис. 3. Ниже приведена общая схема заполнения шаблона с использованием метаинформации.

1. SELECT
 - a. наименование вычисляемого атрибута события
 - b. наименования координатных осей (измерений), в которых вычисляется атрибут
2. FROM
 - a. названия таблиц измерений (поле «Название в базе» таблицы «Измерения»)

b. название таблиц, участвующих в фильтрах (поле «Название в базе» таблицы «Измерения»)

c. название таблицы, содержащей вычисляемый атрибут (поле «Название в базе» таблицы «События»)

3. WHERE

- a. описание связи таблицы событий с таблицами измерений, участвующими в запросе (поле «Текст SQL» таблицы «Измерения событий»)
- b. фильтры по измерениям (поле «Название в базе» таблицы «Измерения» и выбранные значения)
- c. фильтр по атрибуту (поле «Название в базе» таблицы «События» и введенное значение)

4. GROUP BY

- a. названия полей измерений (поле «Название в базе» таблицы «Измерения»)

Интерфейс построителя запросов предоставляет пользователю возможности «разрезания» гиперкуба поперек одного или двух измерений и агрегирования атрибутов, которые характеризуют попавшие в сечения события. Например, для того чтобы построить сечение гиперкуба поперек измерений «группы» и «элементы знаний», надо установить флаги «по горизонтали» и «по вертикали» (если сечение разрезает только одну ось, устанавливается любой из этих флагов) и выбрать названия этих измерений из соответствующих списков. Чтобы сделать предметом анализа среднюю оценку, надо выбрать соответствующий способ агрегирования (другие варианты агрегирования – максимальная или минимальная оценка). Если требуется выполнить усреднение оценки по конкретным группам, надо настроить соответствующий фильтр установкой флага «Группы» и установкой флагов, соответствующих именам групп, задать фильтр по группам.

¹ В терминологии ГИПЕРТЕКСТ элемент знаний – это значимая и независимо оцениваемая процедурами контроля единица учебного материала.

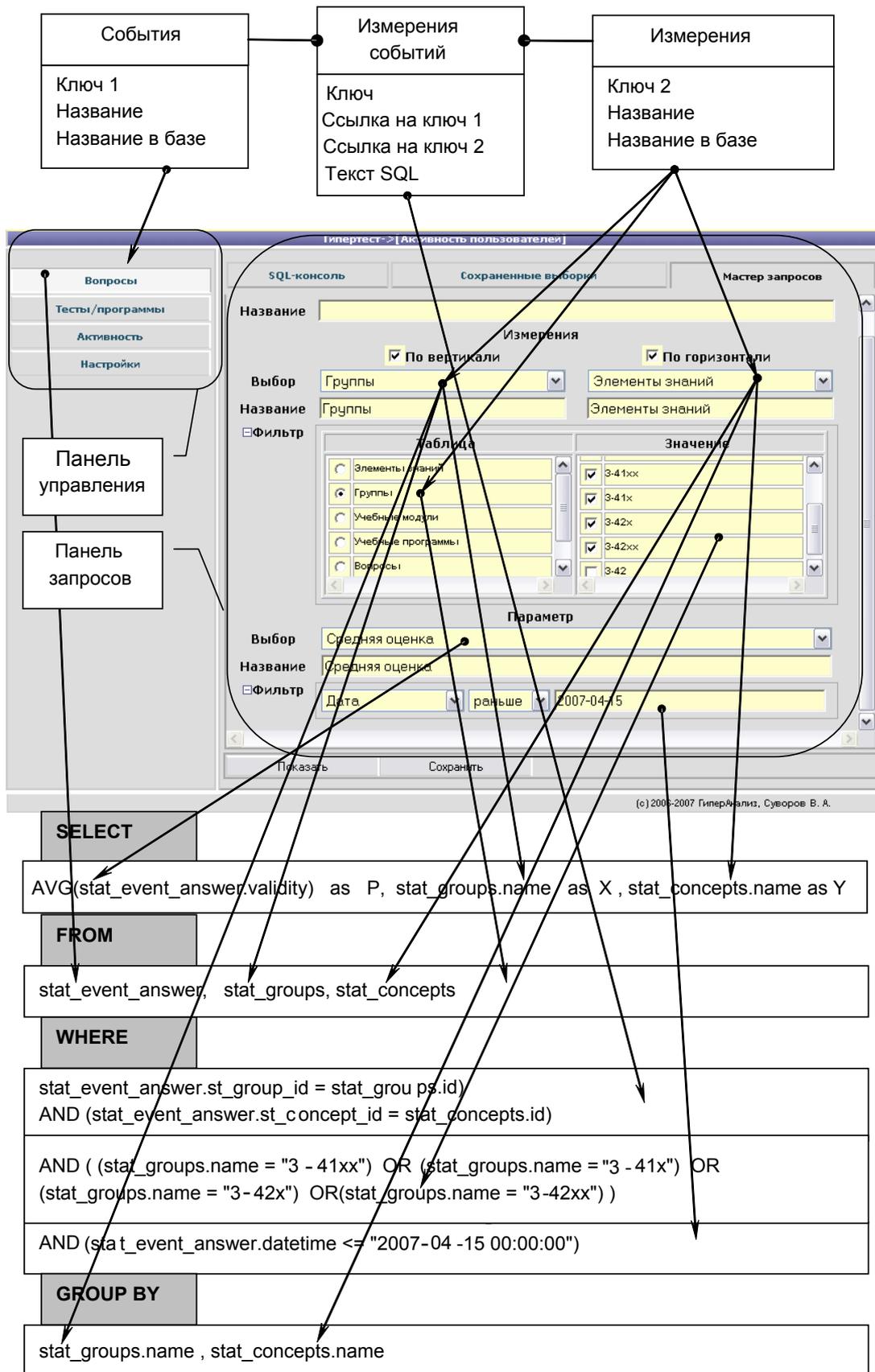


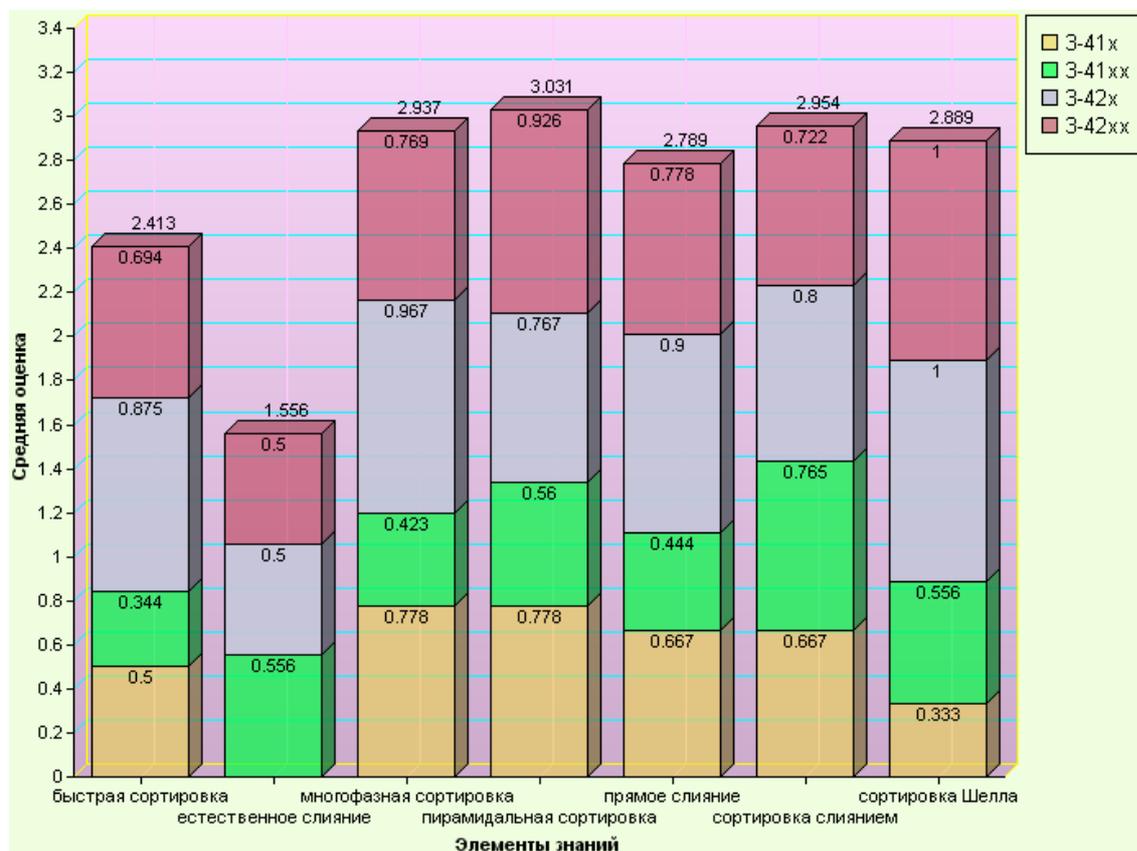
Рис. 3 Использование метаданных для построения запроса к серверу аналитики

Дополнительно может быть определен фильтр по времени наступления событий. Схема взаимодействия метаописания с шаблоном запроса в процессе такой параметризации, а также соответ-

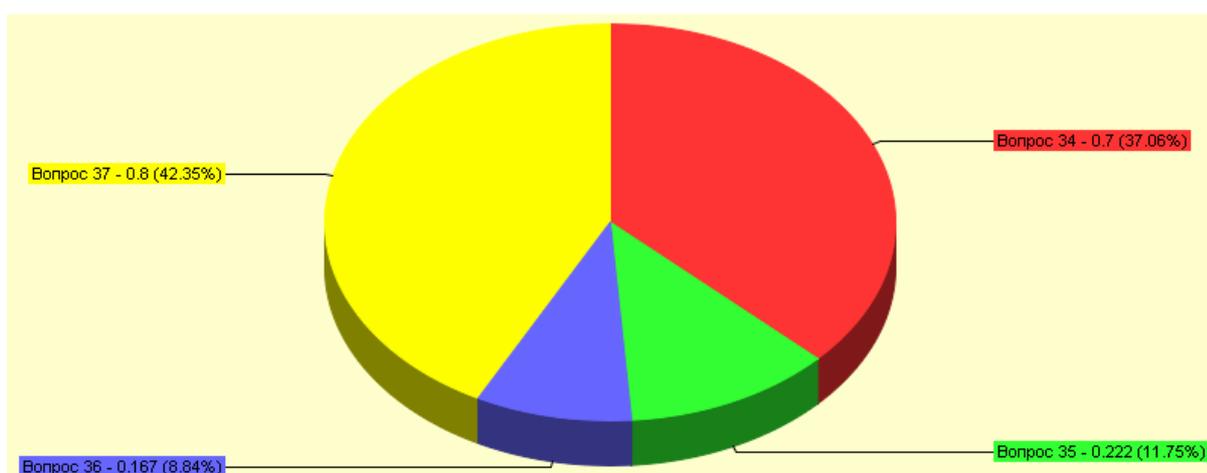
ствующие настройки в интерфейсе построителя запросов показаны на рис. 3. Подготовленный пользователем запрос может быть исполнен (кнопка «Показать») и/или сохранен (кнопка «Сохранить»). Ре-

зультаты исполнения запроса, возвращенные Web-клиенту в виде штабельной диаграммы (stacked bar chart) (рис. 4,а), были получены на основании статистических данных, собранных в процессе обучения студентов ИГЭУ специальности 230105 «Программное обеспечение вычислительной техники и автоматизированных систем» по дисциплине «Структуры и алгоритмы обработки данных» (раздел «Методы сортировки»). Их анализ, в частности, позволил установить, что самые низкие результаты соответ-

ствуют элементу профиля «сортировка естественным слиянием». Так как оценка знаний формируется по результатам ответов на контрольные вопросы, далее был выявлен источник обнаруженного дефекта, т.е. вопрос, вызвавший наибольшие трудности. Анализ среднего балла для каждого из вопросов, оценивающих элемент профиля «сортировка естественным слиянием», показал, что источником дефекта является вопрос № 36 (рис. 4, б).



а)



б)

Рис. 4. Представление результатов выполнения запроса средствами деловой графики: а – результат выполнения запроса пользователя, показанного на рис. 3; б – средние баллы ответа на вопросы, оценивающие элемент профиля «естественное слияние»

SQL-консоль позволяет формулировать любые запросы на выборку данных из хранилища. Составление запроса на языке SQL требует от пользователя знания этого языка и структуры хранилища. Например, чтобы оценить эффективность управления навигацией, выраженную отношением процента активации рекомендованных ссылок к оценке результатов подготовки, необходим запрос-агрегат (рис. 5). Он считает отношения параметров по каждому элементу знаний на выборках, построенных при помощи двух вложенных запросов, которые вычисляют сами значения параметров. Первый вложенный запрос вычисляет количество правильных ответов по каждому элементу знаний. В запросе участвует таблица фактов (`stat_event_answer`) и таблица измерений (`stat_concepts`). Этот запрос возвращает таблицу вида «название элемента знаний»-«количество правильных ответов». Второй вложенный запрос осуществляет расчет количества переходов по ссылкам на каждый из элементов знаний. В запросе участвует таблица фактов (`stat_event_link`) и таблица измерений (`stat_targets`). Этот запрос возвращает таблицу вида «название элемента знаний»-«количество переходов по ссылке». Внешний запрос вычисляет отношения параметров при соответствующих элементах знаний. Результат возвращается в виде таблицы «название элемента знаний (X)»-«значение отношения (P)».

Как запросы, созданные при помощи «Мастера запросов», так и запросы, написанные непосредственно на SQL, могут быть сохранены для повторного использования. Для сохранения запроса на закладках «SQL-консоль» и «Мастер запросов» имеются соответствующие кнопки. Сохраненные запросы доступны на одноименной закладке.

Рассмотренные примеры позволяют утверждать, что результатом реализации предложенного подхода является клиент-серверная архитектура средств оперативной аналитической обработки действий студента в среде Web-обучения, отвечающая заявленным функциональным требованиям (наличие программного интерфейса заполнения и Web-интерфейса обработки и представления данных, пригодного для пользователя с отсутствием навыков работы с OLAP-приложениями). Далее будут рассмотрены технические решения, обеспечивающие открытость сервера аналитики и минимизацию трафика «клиент-сервер».

Под *открытой системой* принято понимать систему, которая обеспечивает возможность совместной работы с локальными и удаленными приложениями благодаря наличию открытых спецификаций на интерфейсы, сервисы и поддерживаемые форматы данных.

Совместная работа сервера аналитики с системами Web-обучения требует от последних поддержки программного интерфейса, построенного на основе библиотеки cURL (сокращение от «Client URLs», [9]). Библиотека cURL предназначена для организации такого взаимодействия серверов, при котором каждый из них может выступать в роли HTTP-клиента. Такое решение позволяет использовать сервис аналитики любому серверу Web-обучения, который поддерживает данный интерфейс. В процессе сбора статистики сервер Web-обучения с помощью набора функций cURL передает серверу аналитики однонаправленный поток данных, характеризующих действия студента.

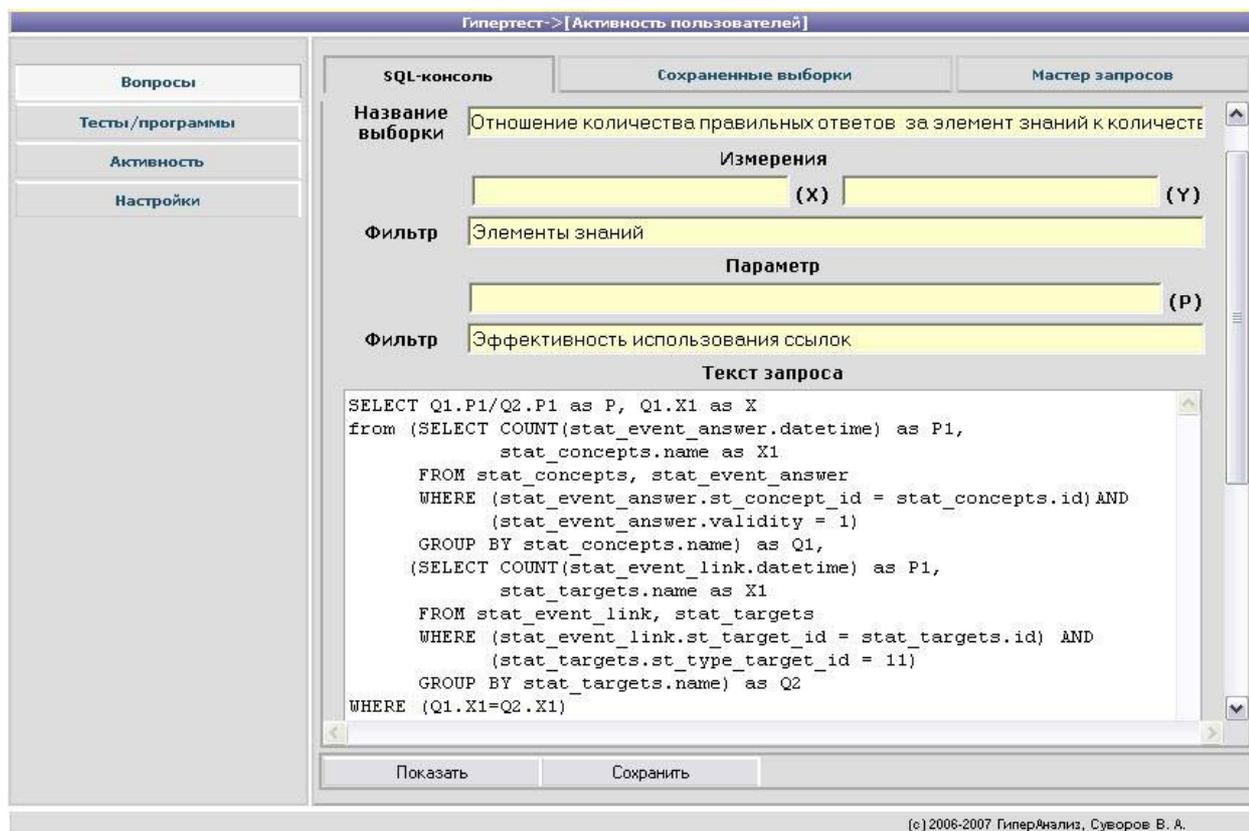


Рис. 5. Редактирование запроса в SQL-консоли

Сценарий передачи данных включает в себя следующие действия:

- инициализацию сеанса связи с сервером аналитики;
- настройку опций сеанса (адрес сервера, метод передачи, поля передаваемых данных);
- передачу информации о событии удаленному серверу аналитики;
- завершение сеанса взаимодействия.

Минимизация трафика «клиент-сервер» в системе Web-OLAP обеспечивается за счет применения технологии Ajax (Asynchronous JavaScript and XML) [10] и серверного компонента деловой графики ChartDirector (<http://www.advsofteng.com/>).

Принципиальное отличие технологии Ajax от традиционного обмена Html-кодом между клиентом и Web-сервером заключается в перераспределении функций обработки информации в пользу клиента [10]. Это достигается за счет размещения на странице Java-скриптов (Ajax-машины), которые включают в себя XMLHttpRequest-объект для взаимодействия с сервером в формате XML и набор функций-обработчиков событий этого объекта (инициализация загрузки данных, соединение с сервером, начало загрузки, завершение загрузки и другие). Посредничество Ajax-машины позволяет существенно уменьшить объем трафика за счет обработки части событий «на месте», а также отправки клиенту только тех данных, которые нужны для обновления страницы. Применение технологии Ajax для создания Web-интерфейса аналитики позволило загружать с сервера только те данные, с которыми в данный момент работает пользователь. Например, информация о сохраненных запросах загружается только после активизации соответствующей закладки.

Компонент ChartDirector предназначен для представления результатов выполнения запросов к хранилищу на странице клиента в формате деловой графики (круговые и столбчатые диаграммы, тренды, радарные схемы и т.п.). Обработка результатов аналитического запроса на стороне сервера также способствует уменьшению трафика «сервер-клиент».

Таким образом, в результате выполненных исследований разработаны клиент-серверная архитектура и базовые компоненты хранилища данных о действиях студента в среде Web-обучения. Хранилище имеет программный интерфейс, который позволяет системам Web-обучения, поддерживающим одноранговый обмен данными в стандарте cURL,

Пантелеев Евгений Рафаилович,
ГОУВПО «Ивановский государственный энергетический университет имени В.И. Ленина»,
доктор технических наук, профессор кафедры программного обеспечения компьютерных систем,
телефон (4932) 26-98-60,
e-mail: erp@poks.ispu.ru

Суворов Виталий Александрович,
ГОУВПО «Ивановский государственный энергетический университет имени В.И. Ленина»,
студент,
телефон (4932) 26-98-60,
e-mail: erp@poks.ispu.ru

заносить в хранилище данные о действиях студента. Хранилище также предлагает Web-интерфейс для построения и оперативного выполнения запросов к хранилищу данных (в том числе, нерегламентированных). Результаты выполнения запросов могут быть представлены в табличном или графическом формате. Для представления деловой графики в интерфейсе Web-клиента использован серверный компонент ChartDirector. Хранилище аналитических данных апробировано на примере организации его взаимодействия с системой Web-обучения ГИПЕРТЕСТ 2.0. Полученные данные были использованы для анализа результатов контроля знания студентов специальности «Программное обеспечение вычислительной техники и автоматизированных систем».

Список литературы

1. **Пантелеев Е.П.** Средства поддержки жизненного цикла Web-обучения в инструментальном комплексе ГИПЕРТЕСТ 2.0 // Информационные технологии. – 2007. – № 2. – С. 39–45.
2. **Codd E.F., Codd S.B., Salley C.T.** Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate.
3. **Baker, R.S.J.d., Corbett, A.T., Koedinger, K.R., Roll, I.** Generalizing Detection of Gaming the System Across a Tutoring Curriculum. In: Proceedings of the 8th International Conference on Intelligent Tutoring Systems. – 2006. – P. 402–411.
4. Mining Student Learning Data to Develop High Level Pedagogic Strategy in a Medical ITS / Michael V. Yudelson, Olga Medvedeva, Elizabeth Legowski, Melissa Castine, Drazen Jukic, Rebecca S. Crowley. In: Proceedings of Workshop on Educational Data Mining at The Twenty-First National Conference on Artificial Intelligence (AAAI 2006), Boston, MA.
5. **Alfred Cobsa.** Generic User Modeling Systems / User Modeling and User-Adapted Interaction. – 2001. – № 11. – P. 49–63.
6. **Brusilovsky P., Sosnovsky S. A., & Shcherbinina O.** (2005). User Modeling in a Distributed E-Learning Architecture. Paper presented at the 10th International Conference on User Modeling (UM 2005), Edinburgh, Scotland, UK, July 24–29, 2005.
7. OLAP-средства и Web-технологии. По материалам зарубежных сайтов. (http://www.ciforum.ru/cfin/articles/olap_web/)
8. Обзор OLAP-продуктов для Web. По материалам зарубежных сайтов. (http://www.ciforum.ru/cfin/articles/olap_web_market/)
9. Владимир Тарасов. Curl (<http://www.ciforum.ru/internet/articles/curl.shtml>)
10. Харви Кейн. AJAX: что скрывается за этими буквами? (<http://www.seonews.ru/article/publication/161/>)