

**Ермаков Е.Ю., Ермаков О.Ю., Паничкина А.А.**

МГТУ им. Н.Э. Баумана, г.Москва, Россия

## **ПРОЕКТИРОВАНИЕ, РЕАЛИЗАЦИЯ И ВНЕДРЕНИЕ В ЭКСПЛУАТАЦИЮ КОМПЛЕКСА СИСТЕМ АВТОМАТИЗИРОВАННОГО МОДЕЛИРОВАНИЯ**

### **АННОТАЦИЯ**

*В статье рассматривается проектирование, реализация и внедрение в учебный процесс веб-приложения "Комплекс Систем Автоматизированного Моделирования". Показана проблематика и задачи, которые решает комплекс. Рассмотрен поэтапный процесс проектирования и создания каждого модуля системы, приведены их архитектурные и функциональные особенности.*

### **КЛЮЧЕВЫЕ СЛОВА**

*КСАМ; Математическое моделирование; Веб-приложение; СУБД; MySQL; MongoDB.*

**Ermakov E.Y., Ermakov O.Y., Panichkina A.A.**

Bauman Moscow State Technical University, Moscow, Russia

## **SERVICE DESIGN, SOFTWARE IMPLEMENTATION AND COMMISSIONING OF WEB APPLICATION "COMPLEX OF AUTOMATED MODELLING SYSTEMS"**

### **ABSTRACT**

*The article describes the design, implementation and introduction in educational process of the web application "Complex Systems Modeling". The paper discusses problems and tasks, which are solved by the complex, examines the gradual process of designing and creating each module of the system, shows their architectural and functional features.*

### **KEYWORDS**

*Complex of automated modelling systems; Mathematical modeling; Web application; DBMS; MySQL; MongoDB.*

### **Введение**

В настоящее время можно наблюдать резко возросший научный интерес к области систем управления баз данных (СУБД). По мнению зарубежных и отечественных ученых в данной области, это связано с назревшим архитектурным кризисом современных СУБД [1]. Ядро крупных СУБД от ключевых производителей (Microsoft, Oracle, IBM) базируются на научных работах Кодда [2] и исследовательских проектах System-R [3] и Ingres [4], созданных в 1970-х. С начала 1980-х гг. основные поставщики СУБД твердо придерживаются стратегии «безразмерности» («one size fits all»), в соответствии с которой поддерживается единственная линия кода для всех служб СУБД [5].

В работах [1,5,6,7,8] некоторые авторы предсказывали конец парадигмы «безразмерности» коммерческих реляционных СУБД. В этих статьях показывается, что производительность основных СУБД в областях хранилищ данных, обработки потоков данных, обработки текстовых данных и научных баз данных может быть превзойдена специализированными программными средствами на один-два порядка величин.

Действительно, сейчас наблюдается «бум» теоретических и научно-практических исследований в области СУБД, направленных на создание узкоспециализированных систем обработки и хранения данных. В свою очередь, это ставит новые задачи перед проектировщиком автоматизированных информационных систем (АИС). Необходимо средство принятия решения, какую СУБД использовать в каждом конкретном случае с учетом специфики предметной области (узкой специализации СУБД) и состава технических и аппаратных средств. Накопленный экспертный опыт в данном случае применить нельзя в силу новизны узкоспециализированных СУБД, а натурные эксперименты могут быть чересчур дорогостоящими. Необходимы средства математического моделирования, позволяющие на этапе проектирования системы предсказать время ее отклика, надежность функционирования и граничные условия применимости различных СУБД.

В следствие этого на кафедре "Системы обработки информации и управления" МГТУ им. Н.Э. Баумана научно-техническим коллективом под руководством д.т.н., профессора Григорьева Ю.А. ведутся теоретические исследования в области построения математических моделей функционирования различных платформ работы с данными: классических реляционных [21,19], массивно параллельных [9,10,11,22], колоночных [12,13,22,20], NoSQL [14,15,17], MapReduce [16,18,19]. Разработанные математические модели достаточно сложны и требуют значительного объема расчетов и приближенных компьютерных вычислений. Более того, для использования разработанных методов необходимо обеспечить возможность хранения и сравнения результатов моделирования двух и более вариантов исследуемой системы, а также вывод информации в графическом виде для более удобной работы аналитика.

Для решения данной проблемы был спроектирован и разработан комплекс средств автоматизированного моделирования (далее КСАМ) распределенных АИС, позволяющий автоматизировать ввод информации о проектируемой системе, расчет ее характеристик и сравнение результатов для различных вариантов моделирования (прямой и обратной задачи исследования проектируемой системы). В случае прямой задачи исследования пользователю КСАМ необходимо анализировать характеристики производительности и выявлять "узкие места" распределенной системы; при этом необходимо определить, какие запросы или транзакции наиболее существенно влияют на загрузки узлов, сетей, время реакции системы. В случае обратной задачи исследования аналитику важно предоставить средства визуального сравнения двух вариантов систем между собой, а также инструментарий для поиска локального оптимума целевой функции проектируемой системы.

### **Подход к проектированию**

Для успешной реализации проекта КСАМ должен быть, прежде всего, адекватно, полно и непротиворечиво описан. При проектировании системы была использована методология объектно-ориентированного анализа и проектирования (OOAD - Object Oriented Analysis and Design) на основе унифицированного языка моделирования (UML - Unified Modeling Language). Такой подход является одним из наиболее распространенных в настоящее время [23,24,25,26], в том числе благодаря полному соответствию концепции объектно-ориентированного программирования (ООП).

Объектно-ориентированное проектирование подразумевает выявление набора сущностей (объектов) и их взаимодействия [27]. Описание системы представляет собой процесс создание ряда моделей, характеризующих различные представления разрабатываемой системы. Полученное множество моделей позволяет получить полное представление как о проектируемой системе в целом, так и об отдельных ее компонентах.

### **Выбор технологий для реализации**

Прототипом создаваемой системы КСАМ 2016-го года является более старая версия КСАМ, описанная в [21]. Основной причиной создания новой версии является требование к охвату проектом большего сектора современных операционных систем для расширения сегмента пользователей, а также возможность использование системы на любой платформе. Следуя из этого требования за основу проекта были выбраны веб-технологии, а именно веб-фреймворк Django, так как веб-версия проекта охватывает все современные операционные системы, в том числе мобильные.

В качестве основной СУБД выбрана реляционная СУБД MySQL [31], базовую интеграцию с которой поддерживает фреймворк Django. Для хранения информации о моделях была выбрана документноориентированная СУБД MongoDB [29], которая хранит информацию в JSON-подобном формате. Данные технологии были выбраны как оптимальные для решения архитектурных проблем при проектировании КСАМ, описанных ниже для каждого модуля в отдельности.

### **Функциональное описание**

Согласно предъявляемым к проектируемой системе требованиям (решение прямой и обратной задачи исследования моделируемой системы), работу с КСАМ можно разделить на 3 функциональные группы (см. рис. 1): ввод информации о моделируемой системе; моделирование и расчет характеристик системы; анализ результатов моделирования.

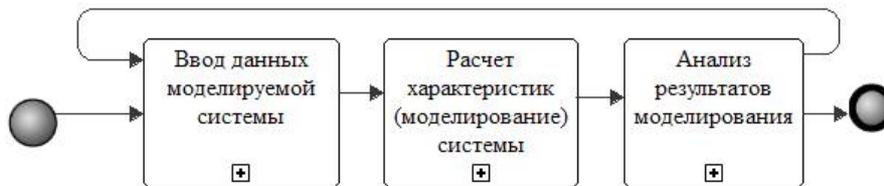


Рисунок 1. Упрощенная последовательность работы с КСМ

Каждый из этих блоков был выделен в отдельный модуль, основные задачи которых описаны ниже.

1. **Задачи модуля ввода информации:** ввод данных о таблицах баз данных проектируемой системы; описание запросов к системе; ввод информации об узлах системы и их характеристиках; описание сетевой архитектуры системы; описание транзакций; описание обращений к транзакциям; распределение транзакций и таблиц БД по узлам системы.

2. **Задачи модуля расчета характеристик:** выбор несколько варьируемых параметров для вычислений; запуск вычислений на сервере веб-приложения; отмена запущенных вычислений; просмотр статуса и процента выполнения вычислений; удаление результатов вычислений с сервера веб-приложения; скачивание результатов вычислений.

3. **Задачи модуля визуализации:** построение графиков по результатам вычислений; скачивание графиков; просмотр и скачивание документации КСМ; просмотр истории действий пользователей, а также пользовательских групп.

### Архитектура

Архитектуру КСМ можно представить в виде ряда взаимосвязанных модулей (см. рис. 2):

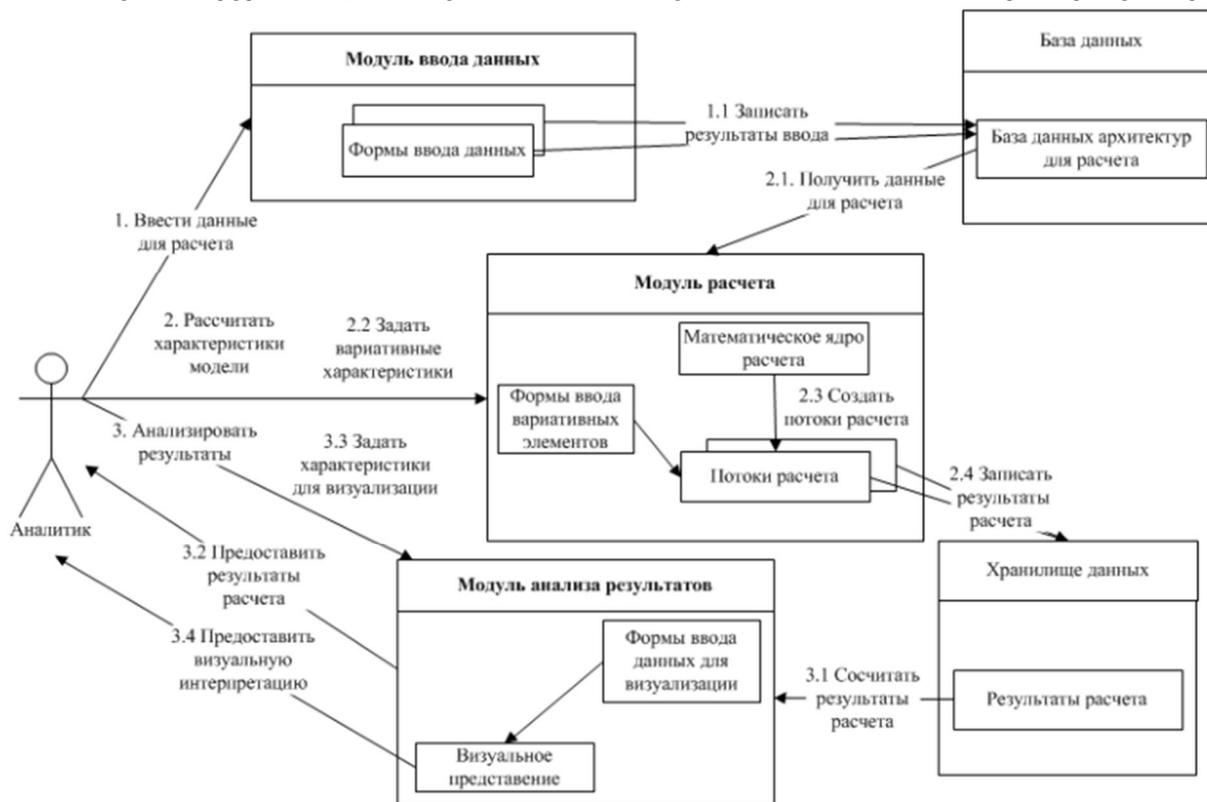


Рисунок 2. Архитектура проектируемой системы

1. Модуль ввод информации о моделируемой системе. Предоставляет пользователю КСМ формы ввода данных о моделируемой системе, который реализован в виде последовательного заполнения одноименных форм пользователем. При вводе данных используется справочная информация из базы данных, пополняемая администратором системы. Полученные данные проходят верификацию и записываются в базу данных для последующего расчета по математической модели в подсистеме моделирования.

2. Модуль моделирования и расчета характеристик системы, который обеспечивает расчет характеристик системы на основе заданных параметров. Аналитик с помощью специальной формы ввода задает вариативные параметры модели. Затем для каждого варианта моделируемой системы запускается поток расчета. Так как каждый из этих расчетов независим друг от друга, весь процесс в целом отлично параллелизуется на независимые потоки. Результаты моделирования каждого варианта записываются в хранилище данных.

3. Модуль визуализации результатов моделирования, который предоставляет пользователю интерфейс для анализа результатов моделирования. С помощью специальных инструментов аналитик может визуализировать зависимости, сравнить несколько вариантов рассчитанной системы, а также выявить «узкие места».

### Особенности модуля ввода информации

Информацию о моделируемой системе можно условно поделить на два крупных блока [21,22]:

1. Концептуальный проект или предметная область, содержащий в себе информацию о структуре базы данных, запросах и транзакциях.

2. Технический проект, содержащий в себе информацию о физических реализациях баз данных, расположенных на кластерах, а также информацию об обращениях к БД из разных узлов.

Взаимосвязи между данными сущностями представлена на рис. 3.

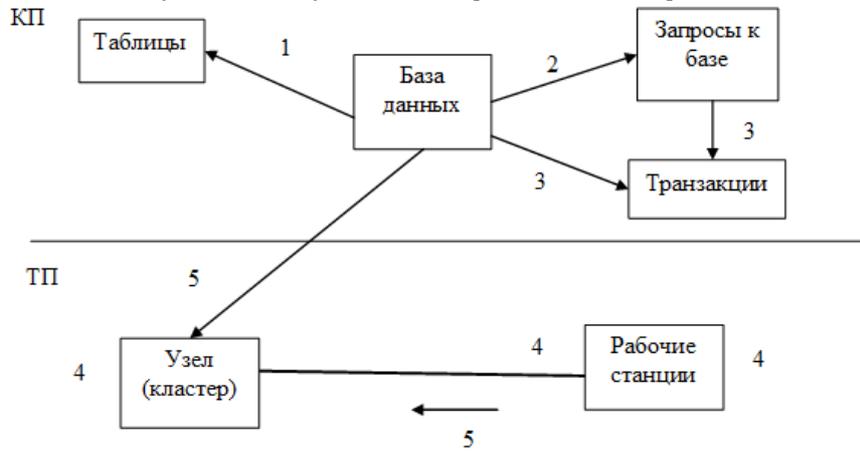


Рисунок 3. Структура вводимой системы

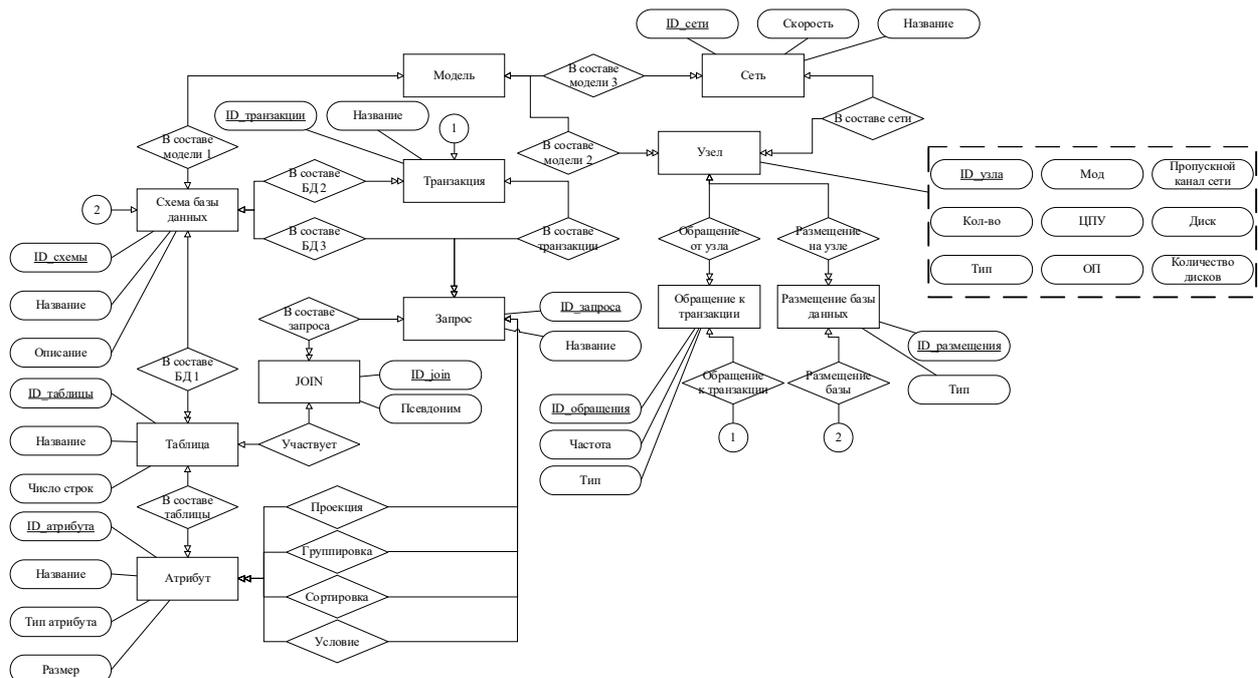


Рисунок 4. Инфологическая модель базы данных модуля ввода

Исходя из структуры, описанной выше, спроектирована инфологическая модель базы

данных модуля ввода, представленная на рис.4.

В рамках поставленной задачи были выявлены следующие особенности, связанные с хранением информации: каждая модель имеет неопределенное число компонентов; поиск по отдельным частям модели не требуется; информация в дальнейшем передается в формате json.

Исходя из выявленных требований хранения информации была выбрана документноориентированная СУБД MongoDB [29], которая хранит информацию в JSON-подобном формате. Преимуществами документноориентированного решения являются [29]: гибкое изменение модели данных, что необходимо для развития проекта и внедрения новых математических моделей; быстрые операции чтения информации, поскольку для того, чтобы достать данные по одной модели требуется осуществить один запрос.

Использование данной СУБД позволило, с одной стороны, повысить производительность и гибкость системы к возможным изменениям структуры данных, а с другой стороны, уменьшить время на проектирование и разработку подсистемы хранения информации.

### Особенности модуля расчета

Спроектированная инфологическая модель модуля расчетов и визуализации представлена на рис. 5.

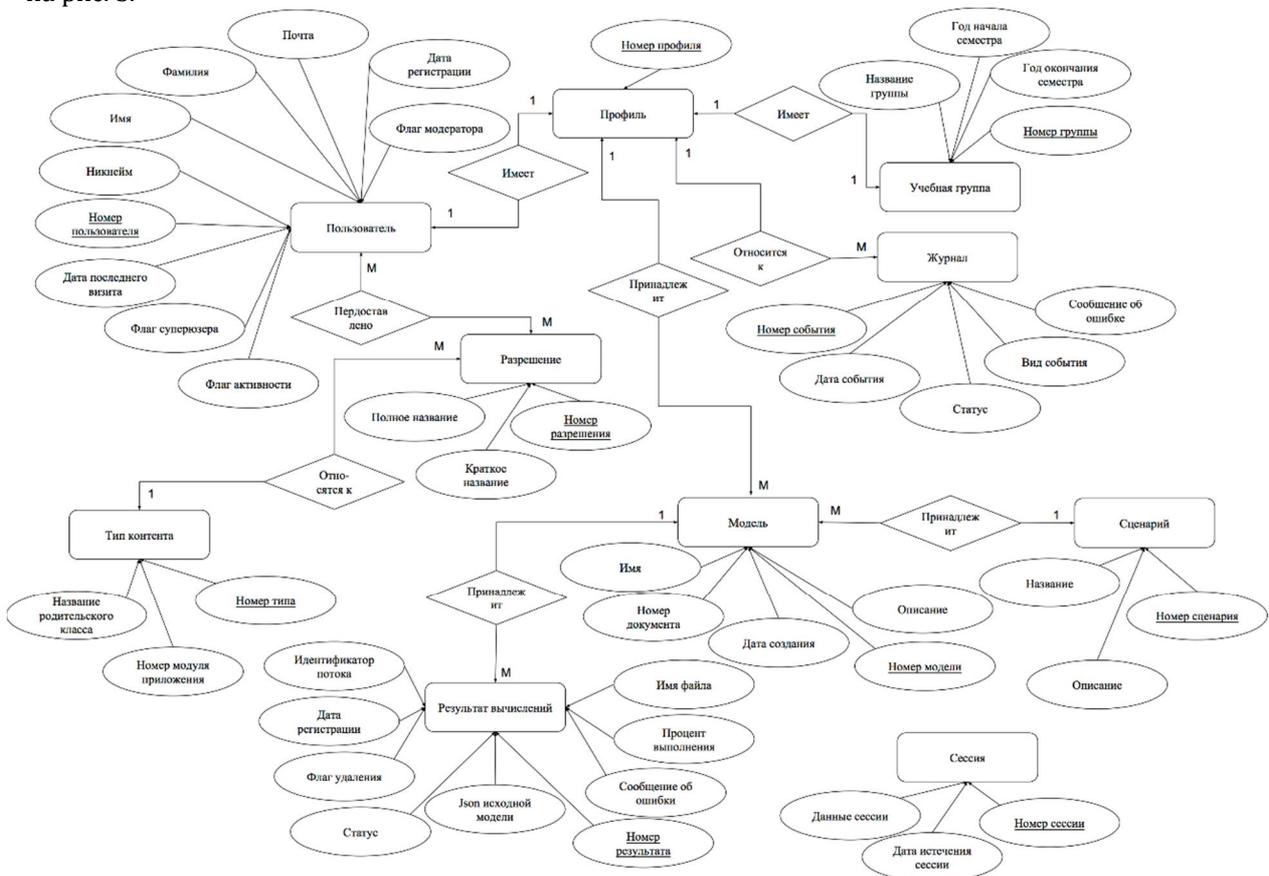


Рисунок 5. Инфологическая модель модуля расчетов и визуализации

Для реализации данной модели выбрана реляционная СУБД MySQL [30], базовую интеграцию с которой поддерживает фреймворк Django.

Даталогическая модель, построенная в MySQL Workbench, представлена на рис. 6.

Основной задачей, решенной при проектировании модуля расчетов KСAM, является реализация возможности использования разработанного комплекса для различных математических моделей, сохраняя общие функциональные возможности ввода, расчета и анализа данных. Общая концепция подобного разделения представлена на рис. 7. Интерфейс математического модуля предоставляет возможность заменять ядро расчетов, регламентируя форматы входных и выходных данных, а также вызываемые функции.

**Входные данные.** Как было показано выше, в качестве входных данных используется json-структура с описанием характеристик моделируемой системы. Перед непосредственно обработкой она проходит валидацию и проверку на корректность введенных значений. После этого происходит загрузка необходимой для расчета модели и считывание параметров для конкретной указанной

СУБД. Так как входные данные включают в себя варьируемые параметры (диапазон и шаг итерации параметра), перед передачей в математическое ядро происходит их преобразование в непосредственно данные для расчета.

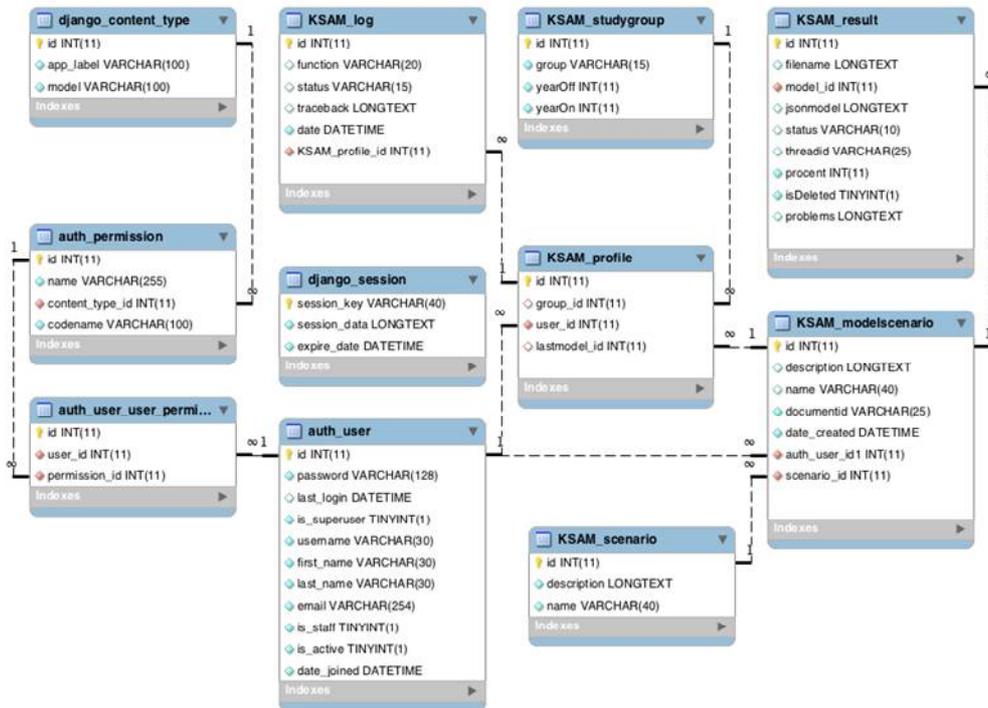


Рисунок 6. Даталогическая модель модуля расчета и визуализации

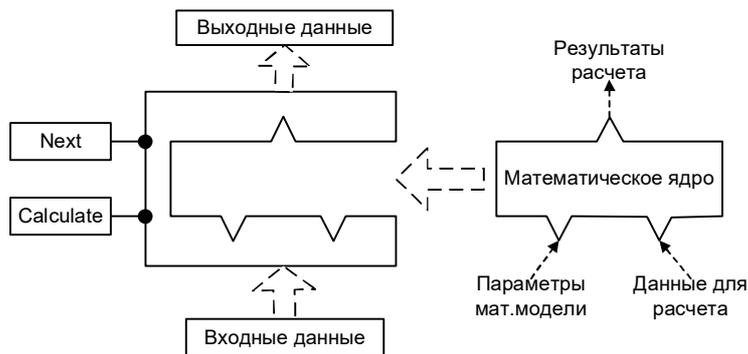


Рисунок 7. Концепция модуля расчета

**Выходные данные.** Представляют собой строки для записи в базу данных KСAM. Содержат в себе как вариативные параметры из входных данных для расчета, так и сами результаты расчета.

**Функции.** В рамках описания интерфейса на верхнем уровне необходимо определить две функции – расчета (Calculate) и считывания следующих параметров для расчета (Next).

**Математическое ядро.** Представляет собой набор алгоритмов, которые на основе данных для расчета и уточняющих коэффициентов выполняют расчет характеристик системы.

Подобную задачу независимости математического ядра от системы можно решить, используя паттерн Мост. Согласно [28], назначение данного паттерна заключается в отделении набора объектов реализаций от набора объектов, использующих их. Данная задача решается посредством следующего набора классов (см. рис. 8)

- Класс Abstraction определяет интерфейс для объектов, представляющих сторону реализации.
- Класс Implementor определяет интерфейс для конкретных классов реализации.
- Классы, производные от класса Abstraction, используют классы, производные от класса Implementor, не интересуясь тем, с каким именно классом ConcreteImplementorX они имеют дело.

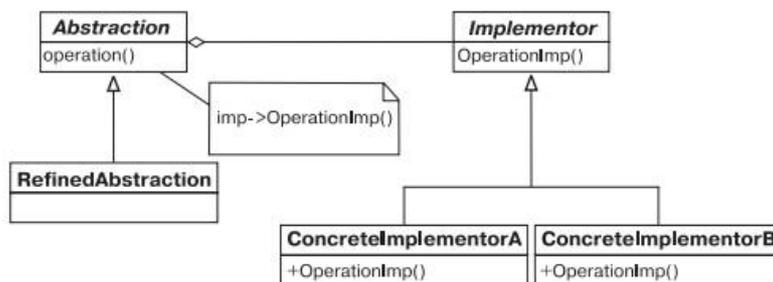


Рисунок 8. Упрощенное представление паттерна Мост [28]

Использование данного подхода позволило создать независимый от реализации интерфейс классов математического ядра, которое стало основой для всех последующих математических моделей, учитывающих особенности своей специализированной СУБД.

### Особенности модуля визуализации

Для создания веб-интерфейса была выбрана библиотека Material Design [31], основными преимуществами которой являются: адаптивность, позволяющая работать с приложением с любых устройств; поддержка большинства современных браузеров; гибкая настройка экранных форм. Использование этой библиотеки позволило транслировать одну версию дизайна на все существующие мобильные платформы.

Разница между мобильной и полной версиями веб-интерфейса демонстрируется на рис. 9.

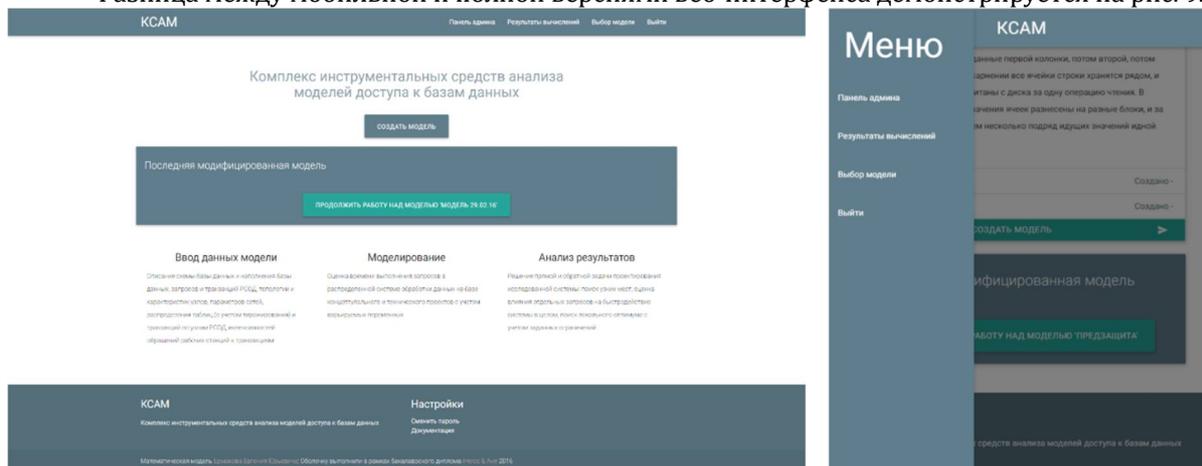


Рисунок 9. (а) Главный экран KSAM

(б) Мобильная версия главного экрана

В мобильной версии приложения (см. рис. 9.б) происходит перенос панели инструментов в боковое меню, а также происходит перестановка элементов на экране.

### Внедрение системы

Спроектированный и разработанный KSAM размещен на удаленном хостинге с доступом по домену <http://ksam.bmstu.ru> и был успешно внедрен в учебный процесс в рамках курса «Структурное проектирование АСОИУ». Данный комплекс стал основой для курсовой работы студентов, демонстрирующей им подходы к проектированию сложных систем и граничные условия работы различных узкоспециализированных СУБД. В научной работе данная система используется на кафедре «Систем обработки информации и управления» МГТУ им Н.Э Баумана как платформа для интеграции собственных математических моделей анализа специфики работы различных СУБД, построения сложных расчетов и анализа их результатов.

### Заключение

В рамках научно-исследовательской работы, проводимой на кафедре «Систем обработки информации и управления» МГТУ им Н.Э Баумана под руководством м.д.т.н., профессора Григорьева Ю.А. был спроектирован, разработан и успешно внедрен в учебный процесс Комплекс Систем Автоматизированного Моделирования. В рамках данной работы:

- создан проект программы на основе унифицированного языка моделирования UML как одного из самых распространённых подходов к проектированию;

- разработана программная реализация системы с использованием фреймворка Django 1.8.3, язык программирования python 2.7;
- для хранения моделей спроектирована база данных на основе нереляционной СУБД mongodb, и разработана структура хранения моделей в формате json, что позволило как уменьшить время разработки данной БД, так и повысить производительность системы в целом;
- разработан веб-интерфейс модуля ввода с использованием библиотеки Materialize CSS; данный выбор предоставил возможность транслировать одну версию дизайна на все существующие мобильные платформы;
- реализована возможность использования разработанного комплекса для различных математических моделей, сохраняя общие функциональные возможности ввода, расчета и анализа данных.

В дальнейшем планируется продолжить работу над улучшением КСАМ, добавив возможность подключения собственных моделей через веб-интерфейс, улучшив возможности визуализации за счет полнофункционального BI-модуля и расширив список языков, на базе которых могут быть реализованы модели.

*Работа выполнена при финансовой поддержке РФФИ в рамках научного проекта 16-37-00117 "Комплекс поддержки принятия решения на этапе проектирования Хранилища Данных"*  
*Руководитель научной работы: профессор, д.т.н. Григорьев Ю.А.*

## Литература

1. Michael Stonebraker, Samuel Madden, Daniel J. Abadi, Stavros Harizopoulos, Nabil Hachem, Pat Helland. The End of an Architectural Era (It's Time for a Complete Rewrite). Proceedings of VLDB, 2007, Vienna, Austria
2. CODD, E.F. A relational model of data for large shared data banks. Comm. ACM IS, 6 (June 1970), 377-387.
3. M. M. Astrahan, Ht. W. Blasgen, D. D. Chamberlin, K. P. Eswaran, J. N. Gray and others System R: Relational Approach to Database Management
4. M. Stonebraker, E. Wong, P. Kreps, and G. Held. The Design and Implementation of INGRES. ACM Trans. Database Systems, 1(3):189-222, 1976.
5. Кузнецов С. Универсальность и специализация: время разбивать камни? Режим доступа: [http://citforum.ru/database/articles/time\\_to\\_break\\_stones/](http://citforum.ru/database/articles/time_to_break_stones/) (дата обращения 10.07.08)
6. Michael Stonebraker, Uğur Çetintemel. «One Size Fits All»: An Idea Whose Time Has Come and Gone.
7. Michael Stonebraker, Chuck Bear, Uğur Çetintemel, Mitch Cherniack, Tingjian Ge, Nabil Hachem, Stavros Harizopoulos, John Lifter, Jennie Rogers, and Stan Zdonik. One Size Fits All? – Part 2: Benchmarking Results, <http://nms.csail.mit.edu/~stavros/pubs/osfa.pdf>.
8. Margo Seltzer. Beyond Relational Databases: There is more to data access than SQL, ACM Queue, Vol. 3, No. 3 - April 2005.
9. Григорьев Ю. А., Плужников В. Л. Оценка времени выполнения запросов и выбор архитектуры параллельной системы баз данных // Информатика и системы управления. – 2009. – № 3.
10. Григорьев Ю. А., Плужников В. Л. Оценка времени соединения таблиц в параллельной системе баз данных // Информатика и системы управления. – 2011. – № 1.
11. Григорьев Ю. А., Плужников В. Л. Анализ времени обработки запросов к хранилищу данных в параллельной системе баз данных // Информатика и системы управления. – 2011.
12. Григорьев Ю. А., Ермаков Е. Ю. Модель обработки запросов в параллельной колоночной системе баз данных // Информатика и системы управления. – 2012. – № 1.
13. Григорьев Ю. А., Ермаков Е. Ю. Модель обработки запроса к одной таблице в параллельной колоночной системе баз данных и анализ ее адекватности // Информатика и системы управления. – 2012. – № 2.
14. Григорьев Ю. А. Анализ свойств баз данных posql // Информатика и системы управления. – 2013. – № 2.
15. Григорьев Ю. А., Цвященко Е. В. Анализ характеристик согласования реплик в конечном счете в базах данных posql // Информатика и системы управления. – 2014. – № 3.
16. Григорьев Ю. А., Пролетарская В. А. Метод ранней материализации доступа к хранилищу данных по технологии MAPREDUCE // Информатика и системы управления. – 2015. – № 3.
17. Григорьев Ю. А., Цвященко Е. В. Анализ надежности баз данных NOSQL // Информатика и системы управления. – 2015. – № 4.
18. Григорьев Ю. А., Пролетарская В. А. Сравнение методов обработки запросов к хранилищу данных по технологии MAPREDUCE // Информатика и системы управления. – 2016. – № 1.
19. Григорьев Ю. А., Устимов А. И. Сравнение времени выполнения запроса к хранилищу данных в среде MAPREDUCE/HADOOP и СУБД MYSQL // Информатика и системы управления. – 2016. – № 3.
20. Ермаков Е.Ю. Операторный метод оценки времени выполнения запросов в параллельной колоночной системе баз данных: дис. ... канд. тех. наук. МГТУ, Москва, 2016.
21. Григорьев Ю.А., Плутенко А.Д. Теоретические основы анализа процессов доступа к распределенным базам данных. Новосибирск: Наука, 2002. - 222 с.
22. Григорьев Ю.А., Плутенко А.Д., Плужников В.Л., Ермаков Е.Ю., Цвященко Е.В., Пролетарская В.А. Теория и практика анализа параллельных систем баз данных. – Владивосток: Даль- наука, 2015.
23. Арлоу Д., Нейштадт А. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование. Второе издание: Пер. с англ.- СПб.: Символ-плюс, 2007.- 624 с.
24. Вендров А.М. CASE-технологии. Современные методы и средства проектирования информационных систем // Citforum.ru. 2008. URL.<http://www.citforum.ru/database/case/index.shtml> (дата обращения 14.07.08)
25. Объектно-ориентированный анализ и проектирование с примерами приложений/ Г. Буч [и др.]: Пер. с англ.- М.: Вильямс, 2008.- 720 с.

26. Овчинников В. Г. Методология проектирования автоматизированных информационных систем. Основы системного подхода. - М.: Компания Спутник +, 2005.- 286 с.
27. Леоненков А.В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. -М.: Бином, 2006.- 320 с.
28. Алан Шаллоуей, Джеймс Р. Тротт Шаблоны проектирования. Новый подход к объектно-ориентированному анализу и проектированию- М.: Издательский дом "Вильямс", 2003.-288с
29. MongoDB. Документ-ориентированная база данных. Режим доступа: <https://www.mongodb.com/> (дата обращения 08.06.2016)
30. Бэнкер К. MongoDB в действии. М.: ДМК Пресс, 2012, 395 с.
31. Официальная документация СУБД MySQL. Режим доступа: <https://www.mysql.com/> (дата обращения 15.06.2016)
32. Materialize. Современная, адаптивная платформа, построенная на принципе материального дизайна. Режим доступа: <http://materializecss.com/> (дата обращения 08.06.2016)

## References

1. Michael Stonebraker, Samuel Madden, Daniel J. Abadi, Stavros Harizopoulos, Nabil Hachem, Pat Helland. The End of an Architectural Era (It's Time for a Complete Rewrite). Proceedings of VLDB, 2007, Vienna, Austria
2. CODD, E.F. A relational model of data for large shared data banks. Comm. ACM IS, 6 (June 1970), 377-387.
3. M. M. Astrahan, Ht. W. Blasgen, D. D. Chamberlin, K. P. Eswaran, J. N. Gray and others System R: Relational Approach to Database Management
4. M. Stonebraker, E. Wong, P. Kreps, and G. Held. The Design and Implementation of INGRES. ACM Trans. Database Systems, 1(3):189-222, 1976.
5. Kuznetsov S. Universal'nost' i spetsializatsiya: vremya razbivat' kamni? Rezhim dostupa: [http://citforum.ru/database/articles/time\\_to\\_break\\_stones/](http://citforum.ru/database/articles/time_to_break_stones/) (data obrashcheniya 10.07.08)
6. Michael Stonebraker, Uğur Çetintemel. «One Size Fits All»: An Idea Whose Time Has Come and Gone.
7. Michael Stonebraker, Chuck Bear, Uğur Çetintemel, Mitch Cherniack, Tingjian Ge, Nabil Hachem, Stavros Harizopoulos, John Lifter, Jennie Rogers, and Stan Zdonik. One Size Fits All? – Part 2: Benchmarking Results, <http://nms.csail.mit.edu/~stavros/pubs/osfa.pdf>.
8. Margo Seltzer. Beyond Relational Databases: There is more to data access than SQL, ACM Queue, Vol. 3, No. 3 - April 2005.
9. Grigor'ev Yu. A., Pluzhnikov V. L. Otsenka vremeni vypolneniya zaprosov i vybor arkhitektury parallel'noy sistemy baz dannykh // Informatika i sistemy upravleniya. – 2009. – № 3.
10. Grigor'ev Yu. A., Pluzhnikov V. L. Otsenka vremeni soedineniya tablits v parallel'noy sisteme baz dannykh // Informatika i sistemy upravleniya. – 2011. – № 1.
11. Grigor'ev Yu. A., Pluzhnikov V. L. Analiz vremeni obrabotki zaprosov k khranilishchu dannykh v parallel'noy sisteme baz dannykh // Informatika i sistemy upravleniya. – 2011.
12. Grigor'ev Yu. A., Ermakov E. Yu. Model' obrabotki zaprosov v parallel'noy kolonochnoy sisteme baz dannykh // Informatika i sistemy upravleniya. – 2012. – № 1.
13. Grigor'ev Yu. A., Ermakov E. Yu. Model' obrabotki zaprosu k odnoy tablitse v parallel'noy kolonochnoy sisteme baz dannykh i analiz ee adekvatnosti // Informatika i sistemy upravleniya. – 2012. – № 2.
14. Grigor'ev Yu. A. Analiz svoystv baz dannykh nosql // Informatika i sistemy upravleniya. – 2013. – № 2.
15. Grigor'ev Yu. A., Tsvyashchenko E. V. Analiz kharakteristik soglasovaniya replik v konechnom schete v bazakh dannykh nosql // Informatika i sistemy upravleniya. – 2014. – № 3.
16. Grigor'ev Yu. A., Proletarskaya V. A. Metod ranney materializatsii dostupa k khranilishchu dannykh po tekhnologii MAPREDUCE // Informatika i sistemy upravleniya. – 2015. – № 3.
17. Grigor'ev Yu. A., Tsvyashchenko E. V. Analiz nadezhnosti baz dannykh NOSQL // Informatika i sistemy upravleniya. – 2015. – № 4.
18. Grigor'ev Yu. A., Proletarskaya V. A. Sravnenie metodov obrabotki zaprosov k khranilishchu dannykh po tekhnologii MAPREDUCE // Informatika i sistemy upravleniya. – 2016. – № 1.
19. Grigor'ev Yu. A., Ustimov A. I. Sravnenie vremeni vypolneniya zaprosu k khranilishchu dannykh v srede MAPREDUCE/HADOOP I SUBD MYSQL // Informatika i sistemy upravleniya. – 2016. – № 3.
20. Ermakov E.Yu. Operatornyy metod otsenki vremeni vypolneniya zaprosov v parallel'noy kolonochnoy sisteme baz dannykh: dis. ... kand. tekhn. nauk. MGTU, Moskva, 2016.
21. Grigor'ev Yu.A., Plutenko A.D. Teoreticheskie osnovy analiza protsessov dostupa k raspredelennym bazam dannykh. Novosibirsk: Nauka, 2002. - 222 s.
22. Grigor'ev Yu.A., Plutenko A.D., Pluzhnikov V.L., Ermakov E.Yu., Tsvyashchenko E.V., Proletarskaya V.A. Teoriya i praktika analiza parallel'nykh sistem baz dannykh. – Vladivostok: Dal'- nauka, 2015.
23. Arlou D., Neyshtadt A. UML 2 i Unifitsirovanny protsess. Prakticheskiy ob"ektno-orientirovanny analiz i proektirovanie. Vtoroe izdanie: Per. s angl.- SPb.: Simvol-plyus, 2007.- 624 s.
24. Vendrov A.M. CASE-tekhnologii. Sovremennyye metody i sredstva proektirovaniya informatsionnykh sistem // Citforum.ru. 2008. URL.<http://www.citforum.ru/database/case/index.shtml> (data obrashcheniya 14.07.08).
25. Ob"ektno-orientirovanny analiz i proektirovanie s primerami prilozheniy/ G. Buch [i dr.]: Per. s angl.- M.: Vil'yams, 2008.- 720 s.
26. Овчинников В. Г. Методология проектирования автоматизированных информационных систем. Основы системного подхода. - М.: Компания Спутник +, 2005.- 286 с.
27. Леоненков А.В. Об"ектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. -М.: Бином, 2006.- 320 с.
28. Алан Шаллоуей, Дзехеймс Р. Тротт Шаблоны проектирования. Новый подход к объектно-ориентированному анализу и проектированию- М.: Издательский дом "Вильямс", 2003.-288с.
29. MongoDB. Документ-ориентированная база данных. Режим доступа: <https://www.mongodb.com/> (дата обращения 08.06.2016)
30. Бенкер К. MongoDB в действии. М.: ДМК Пресс, 2012, 395 с.
31. Официальная документация СУБД MySQL. Режим доступа: <https://www.mysql.com/> (дата обращения 15.06.2016)
32. Materialize. Современная, адаптивная платформа, построенная на принципе материального дизайна. Режим доступа: <http://materializecss.com/> (дата обращения 08.06.2016).

Поступила 11.10.2016

**Об авторах:**

**Ермаков Евгений Юрьевич**, научный сотрудник научно-учебного комплекса “Информатика и системы управления”, кандидат технических наук JK.Ermakov@gmail.com;

**Ермаков Олег Юрьевич**, студент 1 курса магистратуры факультета “Информатика и системы управления”, ihelos.ermakov@gmail.com;

**Паничкина Анна Анатольевна**, студент 1 курса магистратуры факультета “Информатика и системы управления”, annjellyiu5@gmail.com.