

*Г. К. Фаршатов<sup>1\*</sup>, П. Ю. Бугаков<sup>1</sup>*

## **Проектирование и реализация модели базы знаний для экспертной системы в области информационных технологий**

<sup>1</sup> Сибирский государственный университет геосистем и технологий, г. Новосибирск,  
Российская Федерация  
\*e-mail: farshatov-gk2022@sgugit.ru

**Аннотация.** В настоящее время на рынке программного обеспечения представлены единицы проприетарных экспертных систем, разработанных для решения узкоспециализированных задач. Несмотря на высокую эффективность применение экспертных систем в образовании все еще является достаточно редким. В связи с этим разработка собственной экспертной системы для подготовки специалистов в области информационных технологий является актуальной и своевременной. В статье описаны результаты проектирования модели базы знаний, предназначенной для работы в такой экспертной системе. Полученная модель была реализована средствами СУБД Microsoft SQL Server. Проведен контроль структуры базы данных на соответствие первым трем нормальным формам, выполнена настройка и оптимизация быстродействия ее работы с большим объемом данных и высокой интенсивностью запросов. Созданная база знаний будет применяться при разработке экспертной системы для подготовки специалистов в области информационных технологий с использованием нейросетевой модели.

**Ключевые слова:** база знаний, проектирование, база данных, экспертная система, образование

*G. K. Farshatov<sup>1\*</sup>, P. Yu. Bugakov<sup>1</sup>*

## **Design and Development of a Knowledge Base Model for an Expert System in the Field of Information Technology**

<sup>1</sup> Siberian State University of Geosystems and Technologies, Novosibirsk, Russian Federation  
\* e-mail: farshatov-gk2022@sgugit.ru

**Abstract.** Currently, the software market offers a small number of proprietary expert systems designed to solve highly specialized tasks. Despite the high efficiency, the use of expert systems in education is still quite rare. In this regard, the development of our own expert system for training specialists in the field of information technology is relevant and timely. The article describes the results of designing a knowledge base model for work in the expert system. The resulting model was implemented using Microsoft SQL Server DBMS. The control of the database structure for compliance with the first three normal forms was carried out, the configuration and optimization of the speed of its operation with a large amount of data and high query intensity was performed. The created knowledge base will be used in the development of an expert system for training specialists in the field of information technology using a neural network model.

**Keywords:** knowledge base, design, database, expert system, education

### ***Введение***

В настоящее время на рынке программного обеспечения представлены единицы проприетарных экспертных систем, разработанных для решения узкоспециализированных задач. Несмотря на высокую эффективность применение экспертных систем в образовании все еще является достаточно редким [1, 2]. В связи

с этим было принято решение о разработке собственной экспертной системы для подготовки специалистов в области информационных технологий на базе университета. Для осуществления такой разработки необходимо проанализировать принципы функционирования экспертных систем, выявить структуру правил для данной области знаний и на этой основе выполнить проектирование и реализацию модели базы знаний.

### *Методы и материалы*

Для того, чтобы выбрать модель представления данных, которая подходила бы для построения базы знаний, рассмотрим принцип работы экспертной системы на следующем примере. Пользователь вводит элементарный вопрос: «Какое программное обеспечение необходимо использовать для выполнения лабораторной работы 1?» Предположим, ответ системы должен быть «Visual Studio 2022». Учитывая, что разрабатываемая система предполагает использование продукционной модели представления знаний для получения ответа нам необходимо правило: если «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ» и «ЛАБОРАТОРНАЯ РАБОТА» = 1, то «VISUAL STUDIO 2022». При этом в данном правиле используются переменные «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ» со значением NULL и «ЛАБОРАТОРНАЯ РАБОТА» со значением 1.

В качестве модели представления данных используем реляционно-объектный тип представления в силу высокой степени абстракции и распространенности данной модели. В таком случае в соответствии с примером определим два объекта: Правило и Переменная-предикат. Также определим объект для хранения дополнительной информации о переменной-предикате. К примеру, для переменной «VISUAL STUDIO 2022» сохраним описание «линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов» [3].

Поскольку для данной экспертной системы характерно использование большого количества «справочных» запросов, то необходимо использовать такое правило, чтобы при наличии любой подходящей информации в системе выдать ответ. В соответствии с этим положением определим правило: если «ПЕРЕМЕННАЯ», то «ПЕРЕМЕННАЯ». Данное правило гарантирует при наличии декларированной переменной в вопросе пользователя выдать ответ об этой переменной. К примеру, имеем вопрос: «Visual Studio 2022 это?» В соответствии с данным правилом система выведет дополнительную информацию о переменной, описанную ранее.

Поскольку не всегда информация, которую мы хотим описать приходится на единственную переменную определим также объект связи переменных. Данный объект позволит рассматривать несколько переменных как единое целое, для более эффективной работы с переменными без необходимости добавления новых правил. Добавим объект связи в пример с вопросом: «Какое программное обеспечение необходимо использовать для выполнения лабораторной работы 1?» Зададим связь переменных «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ» и «ЛАБОРАТОРНАЯ РАБОТА» = 1. Для реализации данной связи определим в объекте,

хранящем дополнительную информацию, значение-ссылку на переменную «VISUAL STUDIO 2022». Таким образом, в системе не будет необходимости создавать большое количество правил, что упростит добавление новых знаний в систему.

Также немаловажным моментом является необходимость использования переменной-оценки для данной экспертной системы. Эта переменная позволит хранить значение важности, которое может быть получено при помощи нейросетевых технологий. При помощи значения важности система сможет определять какие данные необходимы пользователю при наличии нескольких подходящих переменных. В процессе функционирования система будет обучаться и выдавать пользователям более точные ответы.

### Результаты

Основываясь на результатах анализа принципа функционирования разрабатываемой экспертной системы, была составлена следующая схема взаимодействия таблиц-объектов базы знаний (физическая модель таблиц базы знаний) (рис. 1).

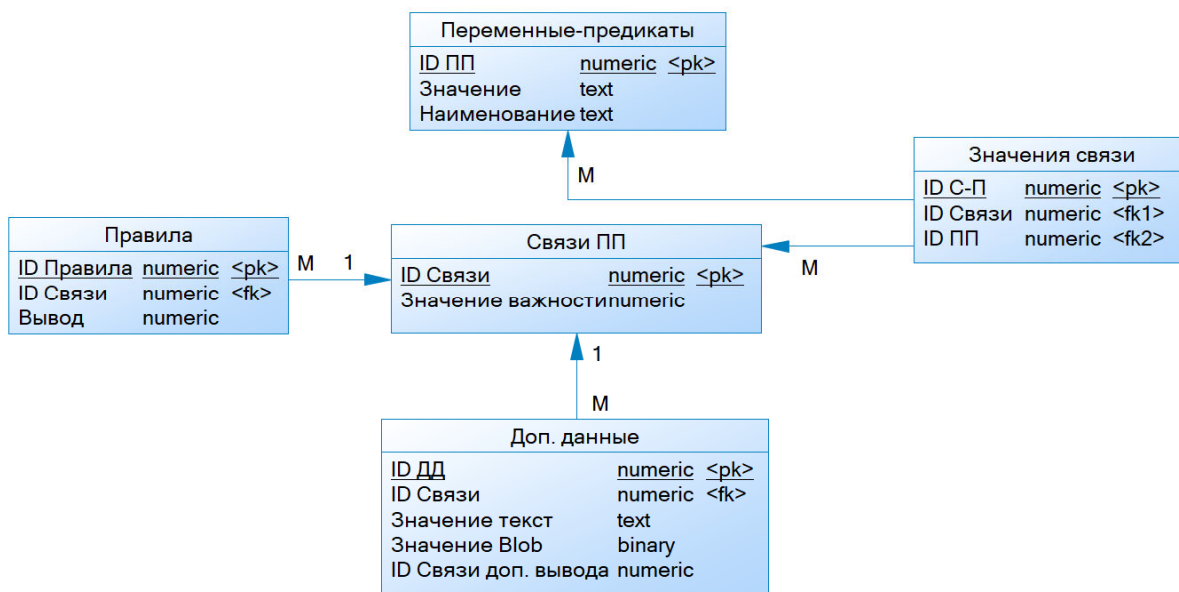


Рис. 1. Физическая модель таблиц БЗ

Для реализации такого функционала экспертной системы, как аутентификация пользователей, ведение журнала действий и хранения кодов ошибок требуется разработать отдельную модель хранения сервисных данных. Аутентификация пользователей требует пять таблиц: роли, пользователи, привилегии, пользователь-роль, роль-привилегия. Для хранения остальных данных потребуется еще три таблицы. В итоге была разработана физическая модель хранения сервисных данных экспертной системы, показанная на рисунке 2.

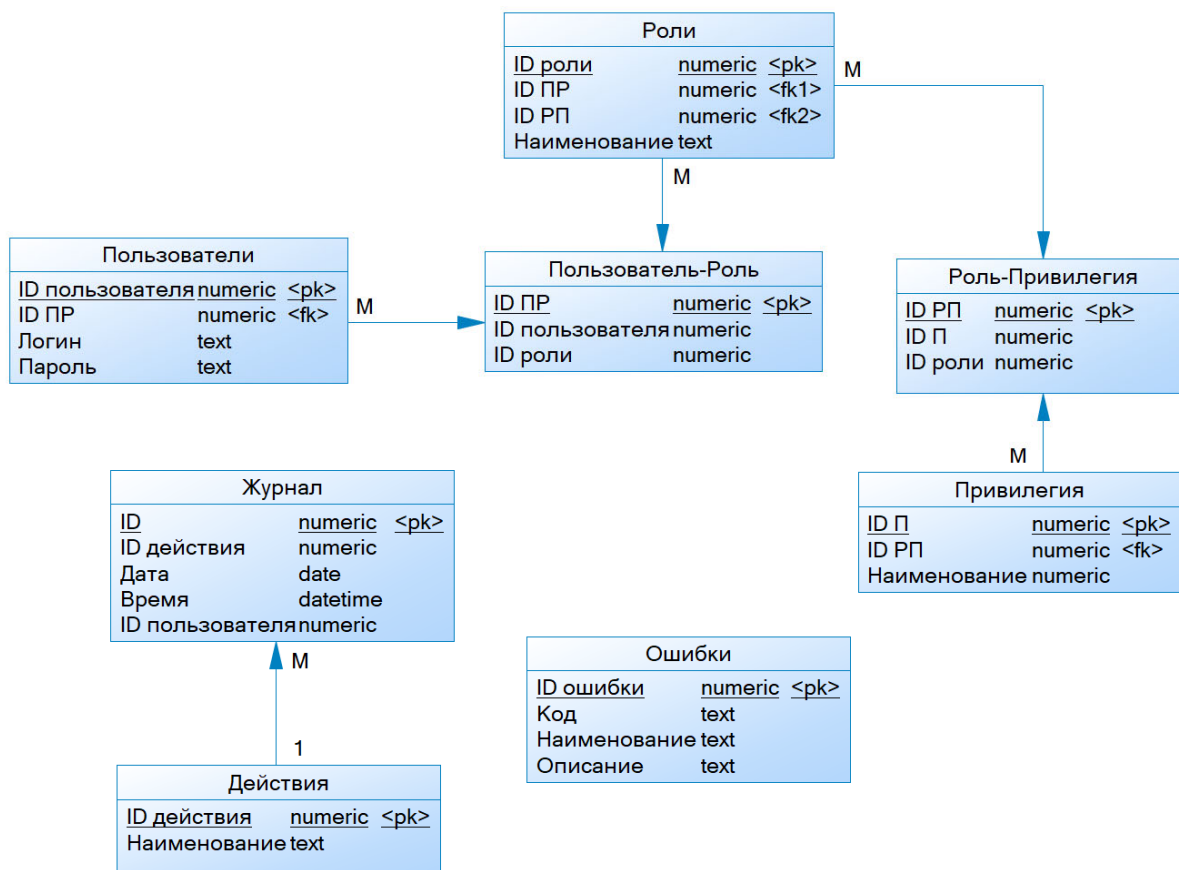


Рис. 2. Физическая модель хранения сервисных данных

Реализация разработанных моделей данных была выполнена средствами СУБД Microsoft SQL Server, которая позволяет создавать хорошо масштабируемые системы с высокой производительностью и достаточно простым администрированием [4]. При выполнении данного этапа разработки были составлены и выполнены SQL запросы на добавление новых таблиц, а также созданы процедуры и функции для добавления и выборки необходимых переменных-предикатов и правил. Использование процедур и функций обусловлено поддержанием целостности данных, а также уменьшением расходов ресурсов на передачу «клиентское приложение – SQL Server».

Для выполнения проверки существования переменной, выборки связей между переменными, добавление записи в журнал, а также проверки аутентификации пользователя также были созданы соответствующие функции и процедуры.

В результате реализации получена база данных со всеми необходимыми средствами для функционирования экспертной системы. Однако создание базы данных на этом не заканчивается. Помимо реализации функциональной части необходимо проверить базу данных на нормализацию и выполнить ее оптимизацию для работы с большим количеством данных в рамках экспертной системы.

База данных считается нормализованной если она соответствует первым трем нормальным формам [5]. Согласно первой нормальной форме, атрибуты таблиц должны быть атомарными. В созданной базе данных указанное правило выполняется. Вторая нормальная форма требует наличие ключа, при этом все неключевые поля таблицы должны зависеть от ключа. Во всех таблицах ключ не составной, а следовательно таблицы находятся во второй нормальной форме. Третья нормальная форма требует отсутствия транзитивной зависимости. Под транзитивной зависимостью понимают отношения, при которых неключевой столбец зависит от значения других неключевых столбцов. Поскольку в качестве ключа используется идентификатор записи, а все значения, которые могут повторятся для разных записей вынесены в отдельные таблицы и связаны через идентификаторы, база данных находится в третьей нормальной форме.

При первом запуске экспертной системы данных будет не много и скорость их обработки будет высокой. Однако, по мере использования системы и ввода новых знаний, данные будут накапливаться и вопрос оптимизации работы базы данных станет критичным. Важным моментом для быстрой работы БД является создание индексов в таблицах. При помощи индексов база данных может существенно ускорять выполнение запросов на выборку. Важно учесть, что создание индекса в таблице приведет к замедлению работы операций вставки, удаления и изменения записи. Наиболее востребованными с точки зрения запросов при использовании системы являются таблицы «Правила», «Переменные-предикаты», «Связи ПП», «Значения связи», «Доп.данные», «Пользователи», «Пользователь-роль», «Роль-Привилегия», «Роли», «Действие», «Ошибки». При использовании базы знаний для всех перечисленных таблиц будет характерно большое количество запросов на выборку и малое на вставку.

В качестве эксперимента реализуем индексы для таблицы «Правила» и проверим результаты выполнения запросов к ней. Для проверки использования индекса получим план запроса до и после добавления индекса (рис. 3, 4).

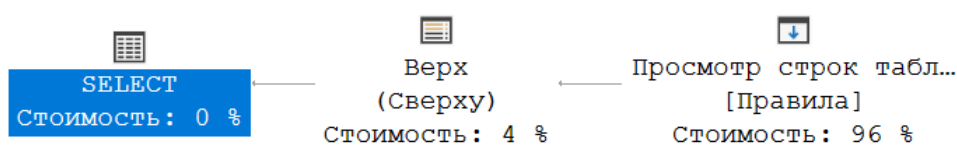


Рис. 3. План запроса без индекса

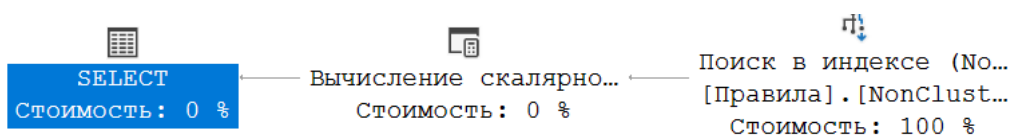


Рис. 4. План запроса с индексом

Оценим разницу во времени при выполнении выборки 1 000 000 записей. Результаты эксперимента получены средствами Microsoft SQL Server Management Studio (табл. 1, 2).

Таблица 1

Результаты скорости выполнения до использования индекса

Номер итерации	1	2	3	4	5	6	7	8	9	10	Сред. значение
Время ожидания при ответе сервера (мс)	94	89	94	90	88	82	82	84	80	84	86,7

Таблица 2

Результаты скорости выполнения с использованием индекса

Номер итерации	1	2	3	4	5	6	7	8	9	10	Сред. значение
Время ожидания при ответе сервера (мс)	0	1	0	0	1	1	0	0	0	1	0,4

Анализируя полученные результаты, можно сделать вывод, что индексирование таблиц принесет существенный прирост производительности экспертной системы на этапе ее активного использования.

После создания индексов для всех оставшихся таблиц необходимо настроить параметры параллелизма. Параллелизм помогает разделять информационно объемные запросы на несколько потоков, выполняемых одновременно. В SQL Server изначально стоит автоматическое определение коэффициента параллельности выполнения запросов. Проблема автоматической настройки параллелизма заключается в использовании всех возможных ресурсов сервера при пиковых нагрузках. Это может приводить к длительному ожиданию обработки запросов пользователей. Помимо этого, SQL Server не всегда корректно определяет необходимость параллельного выполнения запроса, из-за чего возникают дополнительные расходы вычислительных ресурсов на распределение запросов по нескольким потокам. В связи с этим, для многих таблиц использование параллельного выполнения лучше принудительно отключить в настройках.

В то же время для таблиц с большим количеством данных параллелизм может дать существенный прирост в скорости обработки запроса. Такими таблицами в данной базе данных являются: «Журнал», «Правила», «Доп.данные», «Связи ПП», «Переменные-предикаты», «Значения связи». Рекомендуемое значение коэффициента параллелизма равно половине количества ядер сервера [6]. Минимально допустимое количество вычислительных ядер для сервера является 8. Поэтому для перечисленных таблиц базы знаний используем коэффициент параллелизма равный 4.

## *Заключение*

В результате работы была спроектирована модель базы знаний, а также определены необходимые основные данные для работы экспертной системы. Полученная модель была реализована средствами СУБД Microsoft SQL Server. Также проведен контроль структуры базы данных на соответствие первым трем нормальным формам, выполнена настройка и оптимизация ее работы. Полученная база данных обладает всеми необходимыми функциями и процедурами для работы в качестве базы знаний экспертной системы с учетом всех выявленных особенностей.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Молчанов А.А. Использование экспертных систем в системе открытого образования : научная статья // Психолого-педагогический журнал гаудеамус -Тамбов : Тамбовский государственный университет имени Г.Р. Державина, 2014. - 57-68 с. -ISSN: 1810-231X - Текст : непосредственный.
2. Каримова П.М. Применение экспертной системы в образовании // Вестник таджикского национального университета -Москва : Московский институт стали и сплавов, 2020 - 191-195 с. -ISSN: 2074-1847 - Текст : непосредственный..
3. Microsoft Visual Studio [Электронный ресурс] // Wikipedia. – Режим доступа: [https://ru.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio) (дата обращения 17.04.2023).
4. Microsoft sql server: преимущества и недостатки [Электронный ресурс] – Режим доступа: <https://astv.ru/news/materials/microsoft-sql-server-preimushhestva-i-nedostatki> (дата обращения 17.04.2023).
5. Описание нормализации базы данных [Электронный ресурс] // Microsoft – Режим доступа: <https://learn.microsoft.com/ru-RU/office/troubleshoot/access/database-normalization-description> (дата обращения 17.04.2023).
6. Описание нормализации базы данных [Электронный ресурс] // Microsoft – Режим доступа: <https://learn.microsoft.com/ru-ru/sql/database-engine/configure-windows/configure-the-max-degree-of-parallelism-server-configuration-option?view=sql-server-ver16> (дата обращения 17.04.2023).

© Г. К. Фаршатов, П. Ю. Бугаков, 2023