

Зулпукаров Жакшылык Алибаевич – к.ф.-м.н., доцент,
Абдыганы уулу Адилет – магистрант,
Ошский технологический университет

ПОСТРОЕНИЕ ДАТАЛОГИЧЕСКОЙ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ

Разработана структура данных для организации анкетирования студентов. Описан процесс создания структуры базы данных в формате сервера MySQL и созданы механизмы её администрирования через web-интерфейс.

Ключевые слова: Даталогическая модель, концептуальный модуль, реляционная база данных, межзаписной элемент,

Зулпукаров Жакшылык Алибаевич – ф.-м.и.к., доцент,
Абдыганы уулу Адилет – магистрант,
Ош технологиялык университет

ПРЕДМЕТТИК АЙМАКТАРДЫН ДАТАЛОГИЯЛЫК МОДЕЛИ ТҮЗҮҮ

Студенттерди анкетирлөөнү уюштуруу үчүн маалыматтардын структурасы иштелип чыкты. MySQL сервер форматында маалыматтар базасынын структурасын түзүү процессин жазылган жана web-интерфейс аркылуу аны башкаруу механизмдери түзүлдү.

Ачкыч сөздөр: даталогиялык модель, концептуалдык модуль, байланыш маалыматтар базасы, элементтер арасындагы жазуу

Zulpukarov Zhakshylyk Alibaevich, Candidate of physical
and mathematical sciences, docent,
Abdygany uulu Adilet – graduate student,
Osh technological university

CONSTRUCTION OF A DATALOGIC MODEL OF SUBJECT AREA

Developed a data structure for the organization of students' survey. Described the process of creating the database structure in the format of the MySQL server and created the mechanisms for its administration through the web interface.

Keywords: Datalogic model, conceptual module, relational database, inter-element of DBMS.

Проанализировав предметную область и инфологический модульмы прибегаем к построению даталогического модуля реляционной базы данных. Приведем некоторые определения даталогического проектирования.

Даталогический модуль, говоря иначе-логическое проектирование создание схемы базы данных на основе реляционного модуля данных. Даталогический модуль — набор схем отношений, с указанием «связей» между отношениями первичных ключей, а также представляющие собой внешние ключи.

Преобразование концептуального модуля в логический модуль, как правило, осуществляется по формальным правилам. На этапе логического проектирования учитывается конкретный модуль данных, но он может и не учитываться.

Многие СУБД накладывают количественные и иные ограничения на структуру базы данных. Изучение особенностей СУБД обязательно, нужно ознакомиться с существующими методиками проектирования, а также провести анализ имеющихся средств автоматизации проектирования, возможности и целесообразности их использования. [4]

Хотя даталогическое проектирование является проектированием логической структуры базы данных, на него оказывают влияние возможности физической организации данных, предоставляемые конкретной СУБД. Поэтому знание особенностей физической организации данных является полезным при проектировании логической структуры.

Логическая структура базы данных заполняется, отображением реальной предметной области как она есть на самом деле. Поэтому на выбор проектных решений самое непосредственное влияние оказывает специфика отображаемой предметной области, отраженная в инфологическом модуле.

К чему должен приводить результат даталогического проектирования? Конечным результатом даталогического проектирования является описание логической структуры базы данных. Графическое представление используется и при автоматизированном проектировании структуры базы данных как интерфейсное средство общения с проектировщиком, и при документировании проекта. [1]

При проектировании логической структуры базы данных осуществляются все информационные единицы и связи между ними. Следует также задать некоторые количественные характеристики.

Процесс проектирования БД предусматривает предварительную классификацию объектов предметной области, так как в БД отражается определенная предметная область и систематизированное представление информации об объектах и связях между ними.

Соответствующая информация должна быть определено представлена и проанализирована на начальных этапах проектирования БД. В нашем модуле должна быть отображена вся информация, находящаяся и используемая в информационной системе. Одним из первых шагов проектирования является определение состава БД.

При проектировании логической структуры БД происходит преобразование исходной инфологического модуля в модуль данных, которую поддерживает СУБД выбранная нами и проверка адекватности полученной даталогического модуля отображаемой предметной области.

Из множества существующих вариантов проектных решений ее отображения в даталогический модуль мы должны обеспечивать выбор наиболее подходящего проектного решения.

Связи между сущностями предметной области, отраженные в инфологический модуль, могут отображаться в даталогический модуль либо посредством совместного расположения соответствующих им информационных элементов, либо путем объявления связи между ними. Связь может передаваться как на внутрizaписном, так и на межаписном уровне. [3]

Не все виды связей, существующие в предметной области, могут быть непосредственно отображены в конкретный даталогический модуль. Так, многие СУБД не поддерживают непосредственно отношение М:М между элементами. В этом случае в даталогический модуль вводится дополнительный вспомогательный элемент, отображающий эту связь.

Решение о том, какой из способов отображения следует использовать в каждом конкретном случае, будет зависеть от многих факторов. Так, если в предметной области используется классификация вопросов по сложности, в базе данных не следует создавать классификатор сложности, поскольку он будет содержать всего три позиции и никогда не меняется. Как правило, не следует выделять по этому признаку и

соответствующие разделы претметов анкетирования для объектов предметной области, потому что в большинстве случаев они обрабатываются совместно и в основном имеют одинаковый набор свойств, характеризующий их.

При отображении обобщенных объектов в БД возможны разные варианты: хранить информацию обо всем обобщенном объекте в одном файле/таблице, каждому подклассу объектов низшего уровня выделять отдельные самостоятельные файлы/таблицы. Оба эти варианта могут быть использованы в любой СУБД. В этом случае обобщенный объект как единое целое не отображается в структуре базы данных.

При проектировании логической структуры БД основное значение имеет специфика отображаемой предметной области. Однако, как отмечалось выше, и характер обработки информации оказывает влияние на принимаемое проектное решение. Информацию, используемую часто, и информацию, частота обращения к которой мала, также следует хранить в разных файлах, причем последнюю может оказаться выгодным вынести в архивные файлы, а не поддерживать в составе БД. [2]

Для реляционной базы данных проектирование физической структуры заключается в том, чтобы разбить всю информацию по таблицам, а также определить состав полей для каждой из этих таблиц и установить связи между таблицами.

В каждой таблице присутствует уникальное идентификационное поле, которое позволяет однозначно идентифицировать запись таблицы. Такая структура исключает избыточность данных, позволяет ускорить отбор записей по условию, исключает вероятность конфликтов при совместной работе нескольких пользователей.

Одной из основных таблиц является таблица "users". В ней хранится вся информация о сотрудниках, зарегистрированных в системе. Таблица содержит поля - login- идентификационное поле, идентифицирующее единственным образом администратора при входе в систему. Passwd- пароль для доступа к системе тестирования. Пароль кодируется шифром md5, алгоритм которого реализован в языке PHP. Кодирование позволяет защитить пароль от просмотра заинтересованными пользователями. По этим двум параметрам осуществляется вход в систему.

Поле `u_admin` идентифицирует данного пользователя как основного администратора.

Если пользователь является администратором, он обладает дополнительными правами по редактированию базы данных, просмотру статистики, результатов, редактированию вопросов и структуры базы тестов.

Ключевым полем в таблице users является поле login. По полю login данная таблица связана с таблицей "user_courses" связью "один-ко-многим", по полю `abiturient_code`- с таблицей "abiturient_codes" связью "один-к-одному".

В таблице "courses" хранится информация о курсах дистанционного анкетирования. Ключевым полем в таблице является поле `id_course`. Таблица связана по полю `id_course` с таблицей "user_courses" и таблицей "course_parts" связью "один-ко-многим", по полю `n_teacher` с таблицей "teachers" связью "многие-к-одному". Поле `course_period` содержит информацию о продолжительности курса в месяцах. Если абитуриент не прошел тест даты подачи заявки на участие в течение числа месяцев, указанных в `course_period` курс считается не сданным.

В таблице "course_parts" хранится информация о подразделах курсов дистанционного анкетирования. Ключевым полем в таблице является поле `id_course_part`. Таблица связана по полю `id_course_part` с таблицей "questions" и таблицей "user_tests" связью "один-ко-многим", по полю `n_course` с таблицей "courses" связью "многие-к-одному".

В таблице "questions" хранится информация о вопросах анкетирования. Ключевым полем в таблице "questions" является поле `id_question`. Таблица связана по полю `id_question` с таблицей "answers" связью "один-ко-многим", по полю `n_course_part` с таблицей "course_parts" связью "многие-к-одному".

В таблице "user_courses" хранится информация о предметах анкетирования, на которые зарегистрировались студенты. Ключевым полем в таблице "user_courses" является поле id_user_course. Таблица связана по полю id_user_course с таблицей "user_tests" связью "один-ко-многим", по полю login с таблицей "users" связью "многие-к-одному", по полю n_course с таблицей "courses" связью "многие-к-одному".

Шифр абитуриента определяет abiturient_code int.

В таблице "news" хранится информация о новостях анкетирования. Ключевым полем в таблице "news" является поле id_news.

Литература:

1. **Иванова, Г.С.** Основы программирования: Учебник для вузов [Тест] // М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. - 416 с.
 2. **Орлов, С.** Технологии разработки программного обеспечения [Тест] //СПб.:Питер, 2005
 3. **Симонович, С.В.** Практическая информатика [Тест] // Учебное пособие. М.: АСТпресс, 1999.
 4. **Никсон, Р.** Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript. [Тест] // Спб.: Питер, 2011 – 496 с.
-