

Концептуальная, логическая модели и алгоритм проектирования баз данных в доменно-ключевой нормальной форме

© У.А. Тулеев¹, А.А. Алтайбек²

¹Казахский Национальный Университет им. Аль-Фараби, Алматы, Казахстан

²Казкоммерцбанк, Алматы, Казахстан

ualsher.tukeyev@kaznu.kz, aizhan.altaipek@yandex.ru

Аннотация

В докладе рассматривается важность свойства расширяемости баз данных при изменениях условий предметной области. Для решения данной проблемы предлагается упрощенная концептуальная модель «сущность-связь», специальная реляционная модель, использующая только два типа отношений, именно, отношения-сущности и отношения-связывания в доменно-ключевой нормальной форме (ДКНФ), что позволяет упростить процесс проектирования баз данных и повысить степень расширяемости баз данных, что повышает длительность периода эксплуатации базы данных в ее жизненном цикле. Предложенные модели и алгоритм проектирования баз данных показаны на примере проектирования базы данных учебного процесса университета.

Ключевые слова: модель, база данных, проектирование, нормальная, форма, ДКНФ

1. Введение

Длительность жизненного цикла информационных систем, и особенно, длительность периода его эксплуатации, существенно определяются тем, как информационная система удовлетворяет требованиям предметной области. Если требования предметной области изменяются, то приходится производить развитие информационной системы. Ядром информационной системы является база данных (БД), представляющая собой модель предметной области. Любые изменения в предметной области влекут за собой изменения в базе данных информационной системы. Изменения состояния предметной области можно классифицировать на качественные и количественные изменения. Количественные изменения состояния предметной области отражаются в изменениях данных БД, а качественные изменения предметной области отражаются в изменениях структуры БД. Изменения структуры БД могут повлечь за собой изменения

программных модулей системы, что в целом нежелательно для эксплуатируемых систем.

В связи с этим при проектировании информационных систем стоит важная проблема: необходимо так проектировать структуру (модель) БД информационной системы, чтобы минимизировать влияние структурных (качественных) изменений предметной области на информационную систему.

В данной работе предлагаются модели и метод проектирования БД информационных систем с высокой степенью расширяемости структуры БД, основанные на введении специальной реляционной модели, использующей только два типа отношений, именно: «отношение-сущность» и «отношение-связывания», которые должны находиться в доменно-ключевой нормальной форме (ДКНФ).

2. Актуальность свойства расширяемости баз данных

Основными критериями оценки качества модели БД являются «целостность данных», «отсутствие аномалий модификаций» и «отсутствие избыточности данных». Структура БД должна удовлетворять указанным основным критериям качества БД. Если обнаруживается нарушение хотя бы одного из вышеуказанных критериев, то данная структура перепроектируется до тех пор, пока она не будет отвечать указанным требованиям. Внесенные изменения в структуру БД могут принести изменения в прикладных приложениях. Рассмотренные выше проблемы возможных изменений компонентов системы при функционировании информационной системы могут существенно возрасти из-за низкой расширяемости структуры БД.

Определим «расширяемость структуры БД» как количество вносимых модификаций БД без их влияния на существующие отношения БД, т.е. чем

Труды 13^й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL'2011, Воронеж, Россия, 2011.

больше количество модификаций БД без их влияния на существующие отношения БД, тем выше расширяемость структуры БД. И наоборот, чем больше количество вносимых модификаций БД, влияющих на существующие отношения БД, тем ниже расширяемость структуры БД так как, тем больше необходимость внесения изменений в существующие приложения и больше вероятность внесения ошибок в систему. Свойство «расширяемость структуры БД» является очень важным для продления периода эксплуатации информационной системы при будущих изменениях или дополнениях требований предметной области.

3. Критерии проектирования структуры БД

База данных является фундаментальным компонентом информационной системы, а разработку ее структуры (модели) следует рассматривать с точки зрения самых широких требований организации. При разработке модели БД производится ее анализ на соответствие определенным критериям, которым должна удовлетворять построенная модель БД. В данной работе внимание уделяется разработке модели БД, которая удовлетворяет не только основным критериям (целостность, отсутствие аномалий, отсутствие избыточности), но и не менее важному критерию *расширяемости структуры БД*.

Построение модели БД с высокой степенью расширяемости необходимо для БД при изменчивости требований предметной области, для которой была создана эта модель. К сожалению, традиционные модели БД являются эффективными на период разработки системы. Как показывает опыт, модели БД, разработанные таким образом, не подготовлены к новым изменениям или дополнениям предметной области, вызывая ряд проблем, ведущих к перепроектированию структуры БД, а за ним и к реинжинирингу всей информационной системы, что является далеко не желательной процедурой, так как несет за собой немалые разного рода затраты.

При проектировании структуры БД необходимо придерживаться определенных правил, обеспечивающих адекватность модели данных требованиям предметной области. Эти правила обычно называют критериями оценки модели БД, основными которые являются:

- Целостность БД;
- Отсутствие избыточности;
- Отсутствие аномалий.

Проблемы, возникшие после модификации структуры или модели БД, могут быть следующими:

- снижение производительности системы;
- увеличение количества ошибок;
- увеличение вероятности отражения неверных данных;
- нарушение целостности данных;

- появление избыточных данных;
- увеличение физической памяти хранилища данных;
- неверная обработка данных и другие.

Если в результате модификации модели БД, было замечено нарушение одного из основных критериев, то потребуется перепроектирование структуры БД, которая требует нежелательных временных и финансовых затрат, что возможно уменьшить за счет проектирования такой структуры БД, которая обеспечивала бы ее модификацию с меньшими затратами.

Таким образом, предлагается при проектировании структуры БД, наряду с основными критериями учитывать критерий «расширяемость структуры БД», который можно сформулировать также как легкость модификации структуры базы данных без нарушения основных критериев. Здесь легкость модификации структуры БД представляется как выполнение операций включения новых отношений, либо модификация существующих отношений БД без влияния на другие отношения БД. В предлагаемой ниже специальной логической модели это достигается за счет выведения вторичных ключей отношений-сущностей в отдельные отношения-связывания.

4. Анализ существующих методов разработки модели БД

Структура БД, являющейся структурной моделью предметной области, разрабатывается в фазах концептуального, логического и физического проектирования данных. Концептуальное и логическое проектирование базы данных – важные этапы успешности разрабатываемой информационной системы.

Концептуальная модель данных определяет общую схему предметной области, и строится на основе информации, записанной в спецификациях требований пользователя. Создание концептуального представления БД, выполняется посредством определенного метода моделирования. Введенная Питером Ченом (Peter Chen) в 1976 году [1] модель «Сущность-Связь» наиболее популярна для разработки концептуальной модели данных.

В реляционной модели, используемой на этапе логического проектирования, к сожалению, не все отношения одинаково желательны. Для некоторых отношений изменение данных может привести к нежелательным последствиям, называемым аномалиями модификациями. Аномалии могут быть устранены путем разбиения исходного отношения на два или более новых отношений. В большинстве случаев нормализованные отношения являются более предпочтительными.

Э.Ф. Кодд определил следующие нормальные формы отношений: первая нормальная форма (1NF), вторая нормальная форма (2NF), третья нормальная форма (3NF)[2]. Позднее была введена нормальная форма Бойса-Кодда (НФБК), а затем Р. Фагин (R. Fagin) определил четвертую нормальную форму

(4NF) и пятую нормальную форму (5NF)[4,5]. В 1981 году Р.Фагин ввел нормальную форму, которую назвал *доменно-ключевой нормальной формой* (ДКНФ) [6]. В 2002 году была введена шестая нормальная форма (6НФ)[3]. Отношение находится в ДКНФ, если каждое ограничение целостности отношения является следствием ограничений целостности доменов и ключей. В ДКНФ используются три термина: *ограничение*, *ключ* и *домен*. *Ограничение* – это правило, регулирующее возможные статические значения атрибутов и достаточно точное для того, чтобы было установлено, выполняется оно или нет. Как уже было определено, *ключ* – уникальный идентификатор кортежа отношения. *Домен* – это описание допустимых значений атрибута.

Но, как отмечалось ранее в [7, с185] (и до настоящего времени), не известен ни один алгоритм преобразования отношений к ДКНФ, не известно также какие отношения могут быть приведены к ДКНФ. Несмотря на это, при практическом преобразовании баз данных ДКНФ служит в высшей степени полезным ориентиром. Если мы можем вводить отношения таким образом, что все налагаемые на них ограничения являются логическими следствиями доменов и ключей, то в таких отношениях не будет аномалий модификации.

В данной работе предлагаемый метод для разработки логической модели БД, использует упрощенную модель «Сущность-Связь» для построения концептуальной модели данных и ДКНФ при создании логической модели данных.

5. Определение упрощенной модели «сущность-связь»

Концептуальная модель БД является основой для создания логической модели БД. Разработка концептуальной модели осуществляется с помощью упрощенной модели *Сущность-Связь*, где определяются сущности предметной области, их атрибуты и связи между сущностями. В предлагаемой упрощенной концептуальной модели БД не требуется указания детальных связей между сущностями, достаточно просто указать их наличие. Таким образом, упрощенная модель «Сущность-Связь» включает:

- сущности с их атрибутами;
- связи между сущностями без указания их характеристик (атрибутов), т.е. указывается только наличие связи между сущностями (пример упрощенной модели «Сущность-Связь» представлен на рис.4).

6. Специальная реляционная модель БД с высокой степенью расширяемости

Предлагается специальная логическая модель БД, в которой используются только два вида отношений «отношение-сущность» и «отношение-

связывания» ("relation - essence" and "relation - linkages"), и которые должны быть в доменно-ключевой нормальной форме (ДКНФ). Данная модель названа RERL по первым буквам этих отношений на английском языке.

Для предлагаемой специальной логической модели БД будем различать три вида атрибутов:

- *Множество собственных атрибутов* – набор атрибутов, не включающий в себя атрибуты других сущностей, и описывает только одну конкретную сущность;
- *Множество ссылочных атрибутов* – набор атрибутов, состоящий только из идентификаторов сущностей или ключей отношений;
- *Множество дополнительных атрибутов* – набор атрибутов, служащие для отображения дополнительной необходимой информации согласно предметной области.

В предлагаемой специальной логической модели БД введены два вида отношений (рис.1):

- *Отношение-Сущность* – это отношение отражает одну конкретную сущность предметной области и состоит только из множества собственных атрибутов, которое описывает сущность;
- *Отношение-Связывание* – это отношение связывает отношения-сущности между собой и состоит из множества ссылочных атрибутов, являющихся ключами связываемых отношений-сущностей, а также при необходимых требованиях предметной области может иметь множество дополнительных атрибутов.

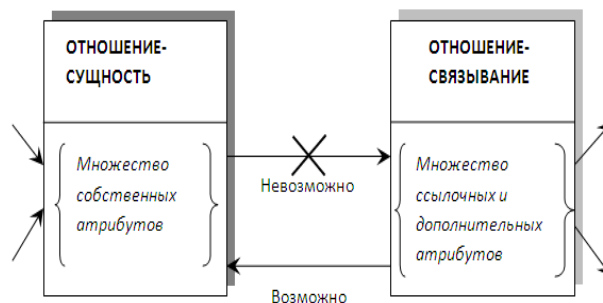


Рисунок 1 - Два вида отношений в логической модели RERL

Введем следующие обозначения. Схемой отношения R называется конечное множество атрибутов $\{A_1, A_2, \dots, A_n\}$. Каждому атрибуту A_i ставится в соответствие множество D_i называемое *доменом* атрибута A_i , $1 \leq i \leq n$. Домен атрибута A_i будем также обозначать как $dom(A_i)$. Пусть $D = D_1 \cup D_2 \cup \dots \cup D_n$. Отношение r со схемой R – это конечное множество отображений $\{t_1, t_2, \dots, t_p\}$ из R в D , где каждое отображение $t \in r$ должно удовлетворять следующему ограничению: $t(A_i)$ принадлежит D_i , $1 \leq i \leq n$. Эти отображения

называются *кортежами*. *A*-значение кортежа t – это конкретное значение кортежа t на атрибуте A , и обозначается как $t(A)$.

Ключ отношения r со схемой R является подмножеством $K = \{B_1, B_2, \dots, B_m\} \subseteq R$, где для любых двух различных кортежей t_1 и t_2 в r существует такое $B \in K$, что $t_1(B) \neq t_2(B)$.

Введем следующие обозначения типов атрибутов:

$P = \{P_1, P_2, \dots, P_k\}$ - множество собственных атрибутов – набор атрибутов, не включающий в себя атрибуты других сущностей, и описывает только одну конкретную сущность;

$C = \{C_1, C_2, \dots, C_m\}$ - множество ссылочных атрибутов – набор атрибутов, состоящий только из идентификаторов сущностей или ключей отношений;

$S = \{S_1, S_2, \dots, S_l\}$ - множество дополнительных атрибутов – набор атрибутов, служащие для отображения дополнительной необходимой информации согласно представлению предметной области.

Определение 1: Отношение–сущность r_e со схемой R_e с множеством атрибутов A_e – это конечное множество отображений $\{t_{e_1}, t_{e_2}, \dots, t_{e_p}\}$ из R_e в D_e , где каждое отображение $t_e \in r_e$ должно удовлетворять ограничению $t_e(A_{e_i})$ принадлежит D_{e_i} , $1 \leq i \leq n$, а множество атрибутов A_e удовлетворять ограничению $A_e = P_e$, где P_e – множество собственных атрибутов для отношения r_e .

Определение 2: Отношение–связывания r_c со схемой R_c с множеством атрибутов A_c – это конечное множество отображений $\{t_{c_1}, t_{c_2}, \dots, t_{c_p}\}$ из R_c в D_c , где каждое отображение $t_c \in r_c$ должно удовлетворять ограничению $t_c(A_{c_i})$ принадлежит D_{c_i} , $1 \leq i \leq m$, а множество атрибутов A_c удовлетворять ограничению $A_c = C_c$ или $A_c = \{C_c, S_c\}$, где C_c – множество ссылочных атрибутов, S_c – множество дополнительных атрибутов для отношения r_c .

По существу *Отношение–связывания* является индексным файлом, представляющим многозначные зависимости между *Отношениями–сущностями*.

В теории реляционных БД используются методы нормализации отношений логической модели БД, основанные на различных типах нормальных форм. Все нормальные формы, кроме

доменно-ключевой нормальной формы (ДКНФ), имеют определенный алгоритм построения модели БД. К сожалению, на данный момент нет алгоритма приведения модели БД к ДКНФ. Предложение в данной работе алгоритма проектирования модели БД в ДКНФ стало возможным благодаря введению двух базовых типов отношений, находящихся в ДКНФ. Это также является основой высокой расширяемости предложенной логической модели, обеспечиваемой за счет выделения вторичных ключей отношений–сущностей в отдельные отношения–связывания. В частности, включение в структуру БД новых отношений с наличием многозначных зависимостей с существующими отношениями в предложенной логической модели не приводят к модификации существующих отношений–сущностей, что повышает расширяемость структуры БД.

7. Алгоритм проектирования БД с высокой степенью расширяемости

Предлагаемый алгоритм проектирования БД включает этапы создания концептуальной и логической моделей БД.

Концептуальная модель БД в данном алгоритме проектирования основана на упрощенной модели *Сущность–Связь*, в которой не указываются детальные связи между сущностями, т.е. связи типа «одним к одному», «один ко многим» или «многие ко многим», достаточно указать только наличие связи.

Алгоритм построения логической модели БД основанный на преобразовании упрощенной концептуальной модели БД в логическую модель RERL состоит из следующих 4 этапов:

1-ЭТАП. Преобразование сущностей в отношения. Все сущности в концептуальной модели «Сущность–Связь» преобразовываются в отношения. Сущности e_1, e_2, \dots, e_n с множеством атрибутов

$$A_{e_1} = \{A_{e_{1_1}}, A_{e_{1_2}}, \dots, A_{e_{1_k}}\},$$

$$A_{e_2} = \{A_{e_{2_1}}, A_{e_{2_2}}, \dots, A_{e_{2_l}}\}, \dots,$$

$$A_{e_n} = \{A_{e_{n_1}}, A_{e_{n_2}}, \dots, A_{e_{n_m}}\}$$
 преобразовываем в

отношения r_1, r_2, \dots, r_n со схемой R , которое соответствует множеству атрибутов

$$A_1 = \{A_{1_1}, A_{1_2}, \dots, A_{1_k}\}, A_2 = \{A_{2_1}, A_{2_2}, \dots, A_{2_l}\}, \dots,$$

$$A_n = \{A_{n_1}, A_{n_2}, \dots, A_{n_m}\}.$$

2-ЭТАП. Декомпозиция отношений на «отношения–сущности» и «отношения–связывания». На этом этапе необходимо проанализировать каждое отношение r , полученное на первом этапе, с целью определения в анализируемом отношении r атрибутов, ссылающихся на другие отношения. Такие ссылочные атрибуты определяем с помощью оператора пересечения схем или атрибутов

отношений. Пусть K будет множеством ключей для отношения $r(A)$ со схемой $R = \{A\} = \{A_1, A_2, \dots, A_l\}$, где A – конечное множество атрибутов, включающее множество ключей $K \subseteq A$. Пусть

$r_i(A_i)$ со схемой

$$R_i = \{A_i\} = \{A_{i_1}, A_{i_2}, \dots, A_{i_j}\}, K_i \subseteq A_i,$$

$r_j(A_j)$ со схемой

$$R_j = \{A_j\} = \{A_{j_1}, A_{j_2}, \dots, A_{j_p}\}, K_j \subseteq A_j$$

будут любыми не одинаковыми отношениями, связь которых видна из модели «Сущность-Связь», тогда декомпозиция выполняется по следующему правилу:

$$r_i(A_i), r_j(A_j), i \neq j, i, j > 0:$$

$$A_i \cap A_j \neq \emptyset, \{A_{i_k}\} = \{A_{j_m}\} \Rightarrow A_i = A_j - \{A_{i_k}\},$$

$$E_{e_i} = A_i', \quad (1)$$

где $1 \leq k \leq l, 1 \leq m \leq p, E_{e_i}$ – множество собственных атрибутов отношения r_i . Другими словами, если в наборе атрибутов A_i отношения r_i существует атрибут $\{A_{i_k}\}$, принадлежащий набору атрибутов A_j отношения r_j , то необходимо удалить атрибут $\{A_{i_k}\}$ из множества атрибутов A_i . В результате мы получаем *отношение-сущность*

$$r_i(A_i) \text{ со схемой } R_i = \{A_i = E_i\},$$

где E_i – множество собственных атрибутов для отношения r_i , и другое *отношение-сущность*

$$r_j(A_j) \text{ со схемой } R_j = \{A_j\},$$

которое изначально содержало множество собственных атрибутов.

Связь между отношениями-сущности r_i и r_j реализуется с помощью введения нового *отношения-связывания*

$$r_c' \text{ со схемой } R_c = \{C_c\} \text{ либо со схемой}$$

$$R_c = \{C_c, S_l\},$$

где C_c – множество ссылочных атрибутов, состоящее из набора ключей $C_c = \{K_i, K_j\}$, а S_l – множество дополнительных атрибутов, наличие которого зависит от конкретных требований.

Таким образом, отношения r_i и r_j были декомпозированы на два отношения-сущности и одно отношение-связывание.

3-ЭТАП. Выполнение ограничений на домены(атрибуты) и ключи. На данном этапе согласно ДКНФ все ограничения (требования) предметной области необходимо представить в качестве логических следствий определения доменов и ключей. Определения ключей, реализованные на втором этапе, покрывает ограничения, относящиеся к ключам, как показано на рисунке 2.

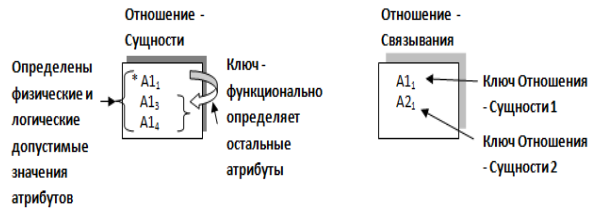


Рисунок 2 – Ограничения на домены и ключи

Ограничения на домены (атрибуты) определяются с помощью их физического и логического (семантического) описания. Физическое описание – это множество значений, которое может принимать атрибут, а логическое описание – это смысл данного атрибута. Ограничения ключей выполняются на уровне СУБД, а ограничения доменов (атрибутов) можно выполнить и на уровне СУБД, и на уровне программного кода, если с помощью СУБД невозможно реализовать требуемое ограничение на домен (атрибут).

4-ЭТАП. Установление связей между отношениями. На этом этапе необходимо реализовать связь между «отношений-сущностей» посредством указания связей с «отношениями-связывания».

В результате выполнения четырех этапов, будет построена логическая модель RERL, представленная в схематичном варианте на рисунке 3. Пунктирные стрелки указывают на возможность связи еще с другими отношениями, не отображенными на схеме. Данная модель будет с высокой степенью расширяемости, и она приведена к ДКНФ, так как все отношения модели RERL находятся в ДКНФ.

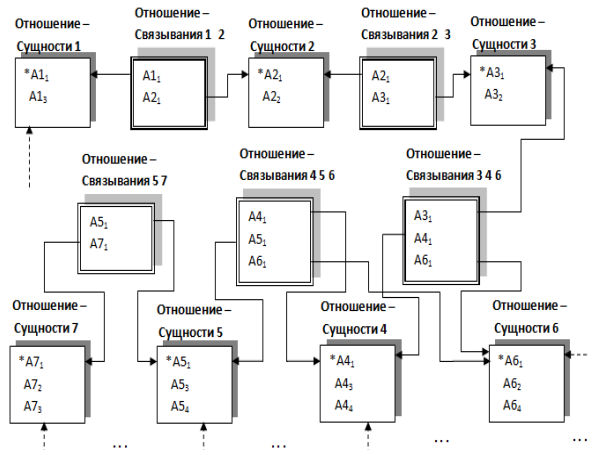


Рисунок 3 – Логическая модель RERL

8 Опыт реализации БД с использованием предложенных моделей и алгоритма проектирования

Реализация предлагаемой модели и алгоритма рассматривается на примере предметной области учебного процесса университета.

Для разработки модели БД необходимо указать требования предметной области, а затем на основе применения предложенного метода выполнить весь процесс проектирования структуры БД.

Определение требований предметной области

Учебный процесс по кредитной технологии состоит из многочисленных требований, правил, задач и функций, требующих автоматизации. В связи с этим рассматриваются следующие процессы кредитной технологии:

- регистрация студентов на дисциплины;
- формирование индивидуального учебного плана (ИУП) студента;
- формирование учебных групп по зарегистрированным дисциплинам;
- ведение БД ППС;
- работа эдвайзеров со студентами;
- ведение электронного журнала посещений и успеваемости.

Для указанных процессов определены следующие требования:

- студенты регистрируются на дисциплины, указанные в компоненте по выбору следующего семестра в определенные сроки;
- для каждой зарегистрированной дисциплины, если количество студентов, зарегистрированных на эту дисциплину более 11, формируют учебную группу;
- студент имеет академическую группу и учебную группу по выбранным дисциплинам;
- ИУП студента формируется на основе выбранных им дисциплин;
- среди ППС назначается эдвайзер;
- эдвайзер назначается на учебные группы;
- ППС может преподавать студентам разных групп, специальностей, курсов, отделения, форм обучения и факультетов;
- ППС может преподавать несколько дисциплин в одном семестре;
- сумма баллов в электронном журнале ППС за каждый рубежный контроль (РК) не должна превышать 30 баллов, а итоговая сумма баллов электронного журнала не должна превышать 60 баллов;
- сумма баллов в электронном журнале ППС должна отображаться в электронной ведомости РК;
- после закрытия электронной ведомости РК, доступ к электронному журналу закрывается.

Разработка концептуальной модели.

Разработку концептуальной модели предметной области, реализуем с помощью упрощенной модели «Сущность-Связь». Рассматриваемая предметная область согласно указанным требованиям имеет следующие сущности:

- e_1 =СТУДЕНТ;
- e_2 =ППС;

- e_3 = ЭДВАЙЗЕР;
- e_4 = СПЕЦИАЛЬНОСТЬ;
- e_5 = ФАКУЛЬТЕТ;
- e_6 = ГРУППА;
- e_7 = ДИСЦИПЛИНА;
- e_8 =ОУП;
- e_9 =ИУП;
- e_{10} = ЭЛЕКТРОННЫЙ ЖУРНАЛ;
- e_{11} = ЭЛЕКТРОННАЯ ВЕДОМОСТЬ;
- e_{12} = МЕСТО ОБУЧЕНИЯ;
- e_{13} = КАФЕДРА.

Определяем атрибуты для каждой сущности. Например, сущность СТУДЕНТ имеет следующие атрибуты:

$A_1 = \{ A_1 = \text{Идентификатор студента}, A_2 = \text{Номер зачетной книжки}, A_3 = \text{Фамилия}, A_4 = \text{Имя}, A_5 = \text{Отчество}, A_6 = \text{Дата рождения}, A_7 = \text{Курс}, A_8 = \text{Код Место обучения}, A_9 = \text{Код Академической группы}, A_{10} = \text{Код группы по выбору}, A_{11} = \text{Код ОУП}, A_{12} = \text{Код ИУП} \}$.

После определения сущностей и их атрибутов, указываем наличие связи между сущностями. Упрощенная концептуальная модель «Сущность-Связь» в рамках указанных требований учебного процесса по кредитной технологии показана на рисунке 4.

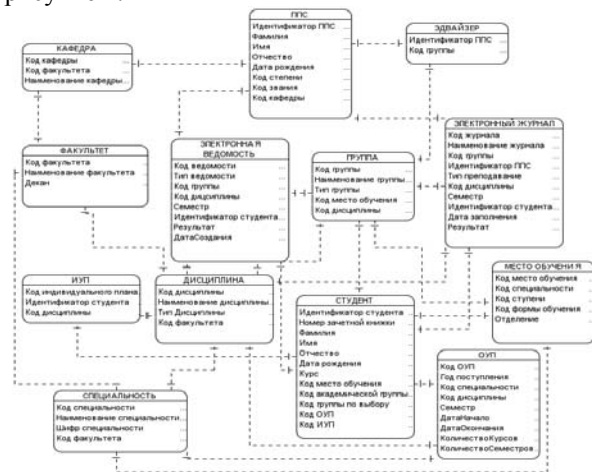


Рисунок 4 – Упрощенная концептуальная модель «Сущность-Связь»

Разработка логической модели БД. Согласно предложенному методу проектирования логической модели БД, создается логическая модель БД путем преобразования концептуальной модели «Сущность-Связь» в логическую модель RERL. Преобразования модели «Сущность-Связь» в модель RERL состоит из четырех этапов:

1-ЭТАП: Преобразование сущностей в отношения. Концептуальная модель имеет сущности $[e_1, e_2, \dots, e_{13}] = \{\text{СТУДЕНТ, ППС, ЭДВАЙЗЕР, СПЕЦИАЛЬНОСТЬ, ФАКУЛЬТЕТ, ГРУППА, ДИСЦИПЛИНА, ОУП, ИУП,}$

ЭЛЕКТРОННЫЙ ЖУРНАЛ, ЭЛЕКТРОННАЯ ВЕДОМОСТЬ, МЕСТО ОБУЧЕНИЕ, КАФЕДРА], которые преобразовываются в отношения. Наименования отношений будут соответствовать наименованию сущностей, а название полей или атрибутов отношений будут соответствовать названию атрибутов.

2-ЭТАП: Декомпозиция отношений на «отношения-сущности» и «отношения-связывания». Декомпозиция отношений выполняется с помощью анализа каждого отношения на наличие атрибутов других сущностей согласно правилу (1) второго этапа.

3-ЭТАП: Формирование ограничений на домены (атрибуты) и ключи. Логические ограничения на ключи выполнены на предыдущем этапе. Ограничения на домены (атрибуты) формируются на основе требований к предметной области, например, что курс студента не может превышать значения 4, или дата начала заполнения электронного журнала не должна начинаться раньше 1 сентября.

4-ЭТАП: Установление связей между отношениями. После того, как уже определены и установлены ограничения на ключи и домены (атрибуты), необходимо установить связи между отношениями. Установление связей между отношениями, также как и предыдущий этап, реализуется на уровне СУБД. Связь реализуется установлением типа связи между соответствующими атрибутами.

Построенная логическая модель БД для учебного процесса по кредитной технологии проиллюстрирована на рисунке 5, где «отношения – сущности» выделены темным цветом, а «отношения – связывания» светлым цветом.

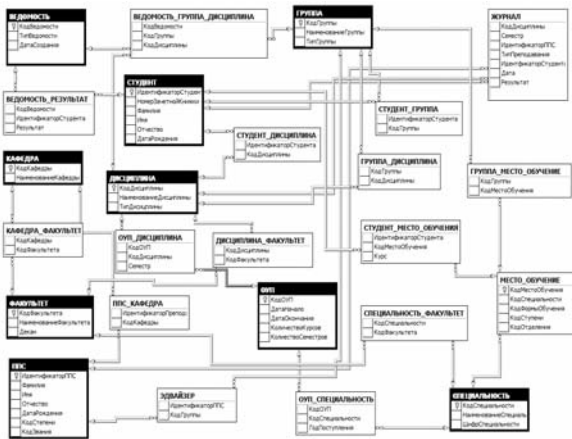


Рисунок 5 – Логическая модель данных БД учебного процесса

9. Заключение

Предложенный метод проектирования модели БД имеет ряд преимуществ по сравнению с традиционными методами проектирования модели БД:

- нет необходимости исследовать модель сущность-связь на наличие разного рода зависимостей (транзитивной, многозначной, зависимости соединения);
- логическая модель состоит только из двух базовых типов отношений, которые строятся в ДКНФ;
- имеется алгоритма проектирования отношений БД в ДКНФ;
- высокая степень расширяемости логической модели БД при качественных изменениях предметной области, достигаемой за счет легкости внесения новых отношений или модификаций старых отношений в структуре БД.

Литература

[1] Chen P.P. The Entity-Relationship Model – Towards a Unified View of Data // ACM Transactions on Database Systems. – 1976. – p. 9-36.
 [2] Codd E.F. A Relational Model of Data for Large Shared Databanks // Communications of the ACM. – Vol.13, №6. – 1970. – p. 377-387.
 [3] Date C.J., Hugh Darwen, Nikos Lorentzos. Temporal Data and the Relational Model. Morgan Kaufmann (2002), p. 176
 [4] Fagin R. (September 1977). "Multivalued Dependencies and a New Normal Form for Relational Databases". ACM Transactions on Database Systems 2 (1): 267.
 [5] Fagin R. "Normal Forms and Relational Database Operators". ACM SIGMOD International Conference on Management of Data, May 31-June 1, 1979, Boston, Mass.
 [6] Fagin R. A Normal Form for Relational Databases That Is Based On Domains and Keys //CACM Transactions on Database Systems, 1981. - Vol.6, №3. - p.387-415.
 [7] Кренке Д. Теория и практика построения баз данных. 8-е изд. – СПб.:Питер, 2003. – 800с.

Conceptual, Logic Models and Algorithm of Designing of Databases in a Domain-Key Normal Form

© U. Tukeyev, A. Altaibek

In the report an importance of property of expansibility of databases is considered at changes of conditions of a subject domain. For solving of the given problem the simplified conceptual model "entity - relationship" model special relational model using only two types of relations 'relation – entity' and 'relation – linkages' in DKNF is offered, that allows to simplify process of designing of databases and to increase a degree of expansibility of databases that enlarges duration of the period of operation of a database in its life cycle. The offered models and algorithm of designing of databases are shown by the example of designing a database of educational process in university.