

## МЕТОД ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ НА ОСНОВЕ ДОМЕННО-КЛЮЧЕВОЙ НОРМАЛЬНОЙ ФОРМЫ (ДКНФ)

У.А. Тукеев, А.А. Алтайбек

*Казахский Национальный Университет имени аль-Фараби*

### **Актуальность проблемы.**

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.

Информационная система, разработанная для конкретной предметной области может прослужить недолгое время, из-за низкой расширяемости структуры БД. Любое изменение предметной области в будущем, требует отражение этих изменений в разработанной информационной системе. Такие модификации могут принести всей информационной системе дополнительные расходы по развитию системы вплоть до разработки новой системы.

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

Главной частью любой информационной системы является ее БД, где хранятся и обрабатываются все данные системы. БД являясь ядром системы, играет основную роль в работе информационной системы. Создание структуры БД – это создание модели предметной области. Для построения такой модели существуют критерии, которым должна удовлетворять построенная модель. Основными критериями оценки модели БД являются «целостность данных», «отсутствие аномалий модификаций» и «отсутствие избыточности данных». Структура БД должна удовлетворять как минимум указанным основным критериям. Если обнаруживается нарушение хотя бы одного из вышеуказанных критериев, то данная структура отклоняется, либо перепроектируется до тех пор, пока она не будет отвечать указанным требованиям. Со временем любая предметная область изменяет или дополняет свои требования, следовательно, в разработанную для исходных требований структуру БД, теперь уже не отвечающую новым требованиям, необходимо будет вносить изменения. Внесенные изменения в структуру БД после ее окончательной разработки, приводят к нежелательным проблемам, а удовлетворения основных критериев могут нарушаться. Так как, БД является фундаментальным компонентом всей информационной системы, то из-

менение, внесенные в структуру БД отражаются и на других компонентах информационной системы. После таких изменений в информационной систем возможны следующие проблемы:

- снижение производительности системы;
- увеличение вероятности отражения неверных данных;
- нарушение целостности данных;
- появление избыточных данных;
- увеличение памяти хранилища данных;
- неверная обработка данных и т.д.

Если отмеченные проблемы невозможно устранить или нарушается один из основных критериев оценки структуры БД, то требуется переработка или перепроектирование структуры БД. Следовательно, это ведет к переработке всей информационной системы, что является для заказчиков или пользователей системы весьма нежелательной проблемой, так как влечет немало временных и финансовых затрат.

Выявленные выше недостатки функционирования информационной системы возникают из-за низкой расширяемости структуры БД. Данная ситуация показывает, что «расширяемость структуры БД» является очень важным критерием для продления периода эксплуатации информационной системы, несмотря на будущие изменения или дополнения требований предметной области.

Отсюда следует, что важным критерием проектирования БД является критерий расширяемости структуры БД, который должен обеспечить реализацию качественных изменений предметной области в информационной системе без изменения уже существующих модулей системы.

В данной работе предлагается метод проектирования БД информационных систем, основанный на введении логической реляционной модели, использующей только два типа отношений, именно: «отношение-сущность» и «отношение-связывания», которые должны находиться в доменно-ключевой нормальной форме (ДКНФ).

### **Метод проектирования БД.**

В теории реляционной БД используются методы нормализации для создания логической модели БД, основанные на различных типах нормальных форм. Все нормальные формы, кроме доменно-ключевой нормальной формы (ДКНФ), имеют определенный алгоритм построения модели БД, ориентированный только на удовлетворения основным трем критериям оценки модели БД. ДКНФ являясь самой поздней и совершенной из всех нормальных форм, согласно определению не имеет каких-либо аномалий и недостатков/1/. К сожалению, на данный момент нет алгоритма приведения модели БД к ДКНФ. Следовательно, вопрос создания метода (алгоритма) построения модели БД в ДКНФ и имеющей высокую степень расширяемости, является актуальным и имеет высокую теоретическую и научную ценность.

В данной работе предлагается метод проектирования БД, имеющий следующие особенности:

- 1) для проектирования логической модели БД предлагается использовать упрощенную концептуальную модель «Сущность-Связь». Упрощенная концептуальная модель «Сущность-Связь» характеризуется тем, что в ней указывается только то, что сущности связаны между собой, но не указываются тип связи, а именно, каким образом связаны сущности между собой, то есть, нет необходимости указывать связи типа «одним к одному», «один ко многим» или «многие

ко многим». Главное показать, какие сущности связаны между собой.

2) Предлагается использование логической модели БД, в которой используются только два вида отношений «отношение-сущность» и «отношение-связывания», и которые должны быть в доменно-ключевой нормальной форме (ДКНФ). Данная модель названа ОС2 по первым буквам этих отношений.

3) При построении логической модели БД атрибуты сгруппированы на три вида:

- $P = \{P_1, P_2, \dots, P_k\}$  - 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 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  $r_c$ .

Согласно модели упрощенной модели «Сущность-Связь», объекты представляются сущностями, а взаимоотношения между этими сущностями представляются явно определенными связями. Данная модель должна определить и предоставить все сущности  $e_1, e_2, \dots, e_n$  предметной области, с указанием множества атрибутов  $A_{e_1} = \{A_{e_{11}}, A_{e_{12}}, \dots, A_{e_{1k}}\}$ ,  $A_{e_2} = \{A_{e_{21}}, A_{e_{22}}, \dots, A_{e_{2l}}\}, \dots, A_{e_n} = \{A_{e_{n1}}, A_{e_{n2}}, \dots, A_{e_{nm}}\}$  для каждой из этих сущностей, а также, связи  $c_1, c_2, \dots, c_p$  между указанными сущностями. В данном случае нет необходимости указывать, как именно связаны сущности, достаточно просто определить их наличие между этими сущностями. Когда модель «Сущность-Связь» готова приступаем к процессу преобразования ее в модель ОС2, состоящий из следующих четырех этапов.

Метод построения логической модели БД основан на упрощенной концептуальной модели БД, и состоящий из следующих 4 этапов:

**1-ЭТАП преобразование сущностей в отношения.** Все определенные сущности в концептуальной модели «Сущность-Связь» преобразовываем один к одному в отношения. В процессе преобразования необходимо обратить внимание на идентификаторы сущностей, так как они будут ключами в преобразованных отношениях. Необходимо определить ключи для всех преобразованных отношений. Если

идентификатора у сущности нет, тогда необходимо задать его в отношении в виде ключа.

Сущности  $e_1, e_2, \dots, e_n$  с множеством атрибутов  $A_{e_1} = \{A_{e_{11}}, A_{e_{12}}, \dots, A_{e_{1k}}\}$ ,  $A_{e_2} = \{A_{e_{21}}, A_{e_{22}}, \dots, A_{e_{2l}}\}$ , ...,  $A_{e_n} = \{A_{e_{n1}}, A_{e_{n2}}, \dots, A_{e_{nm}}\}$  преобразовываем в отношения  $r_1, r_2, \dots, r_n$  со схемой  $R$ , которое соответствует множеству атрибутов  $A_1 = \{A_{11}, A_{12}, \dots, A_{1k}\}$ ,  $A_2 = \{A_{21}, A_{22}, \dots, A_{2l}\}$ , ...,  $A_n = \{A_{n1}, A_{n2}, \dots, A_{nm}\}$ .

На данном этапе реализуется только процесс преобразования сущностей в отношения, и установления ключей в этих отношениях. На этом первый этап завершается, после чего переходим к следующему этапу.

**2-ЭТАП декомпозиция отношений на «отношения-сущности» и «отношения-связывания».** На этом этапе необходимо проанализировать каждое отношение  $r$ , полученное на первом этапе. Главная задача - определить в анализируемом отношении  $r$  атрибуты, ссылающиеся 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

$$\begin{aligned} r_i(A_i) \text{ со схемой } R_i = \{A_i\} = \{A_{i_1}, A_{i_2}, \dots, A_{i_l}\}, K_i \subseteq A_i, \\ r_j(A_j) \text{ со схемой } R_j = \{A_j\} = \{A_{j_1}, A_{j_2}, \dots, A_{j_p}\}, K_j \subseteq A_j \end{aligned}$$

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

$$r_i(A_i), r_j(A_j), i \neq j, i, j > 0: A_i \cap A_j \neq \emptyset, \{A_{i_k}\} = \{A_{j_m}\} \Rightarrow A'_i = A_i - \{A_{i_k}\}, E_{e_i} = A'_i, \quad (1)$$

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

$S_i$  – множество дополнительных атрибутов, наличие которого зависит от конкретных требований.

Следовательно, отношения  $r_i$  и  $r_j$  были декомпозированы на два отношения-сущности и одно отношение-связывание. Здесь необходимо добавить, что некоторые отношения в рамках требований предметной области могут состоять только из атрибутов других сущностей. Данное множество атрибутов, хотя и содержит множество атрибутов других сущностей, оно будет собственным, так как описывает конкретную сущность. Следовательно, декомпозицию необходимо выполнять, учитывая требования предметной области.

Описанный процесс декомпозиции является общей схемой необходимой декомпозиции, хотя на самом деле данный процесс может быть более сложным в зависимости от концептуальной модели «Сущность-Связь», и наличие множества дополнительных атрибутов в отношении-связывание зависит от требований предметной области.

**3-ЭТАП выполнение ограничений на домены 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 1.

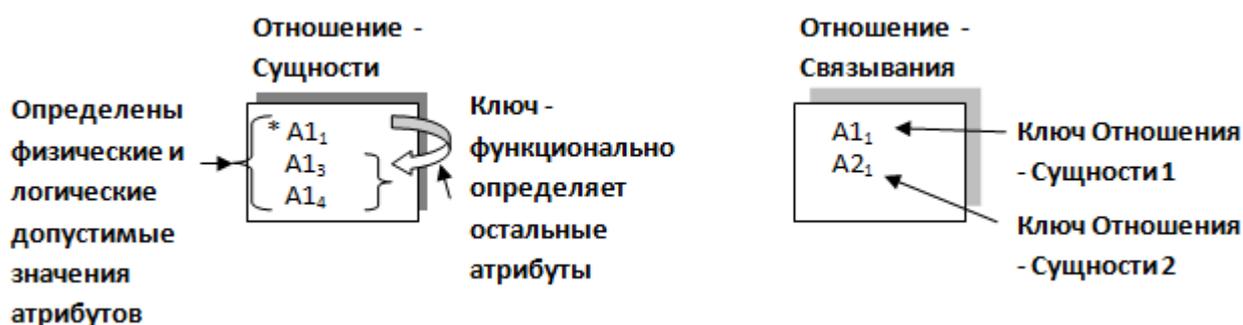


Рисунок 1 – Ограничения на домены и ключи

Следовательно, логическое следствие ключа уже представлено. Ограничения на домены 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 можно выполнить и на уровне СУБД, и на уровне программного кода, если с помощью СУБД невозможно реализовать требуемое ограничение на домен.

**4-ЭТАП установление связей между всеми отношениями.** Данный этап является завершающим этапом предлагаемой методики, где устанавливаются связи между всеми отношениями. На этом этапе необходимо реализовать связь между «отношениями-сущностями» посредством указания связей с «отношениями-связывания».

В результате выполнения четырех этапов, будет построена логическая модель ОС2, представленная в примерном схематичном варианте на рисунке 2. Пунктирные стрелки указывают на возможность связи еще с другими отношениями, не отобра-

женных на схеме. Данная модель будет с высокой степенью расширяемости, и она приведена к ДКНФ, так как все отношения модели ОС2 находятся в ДКНФ.

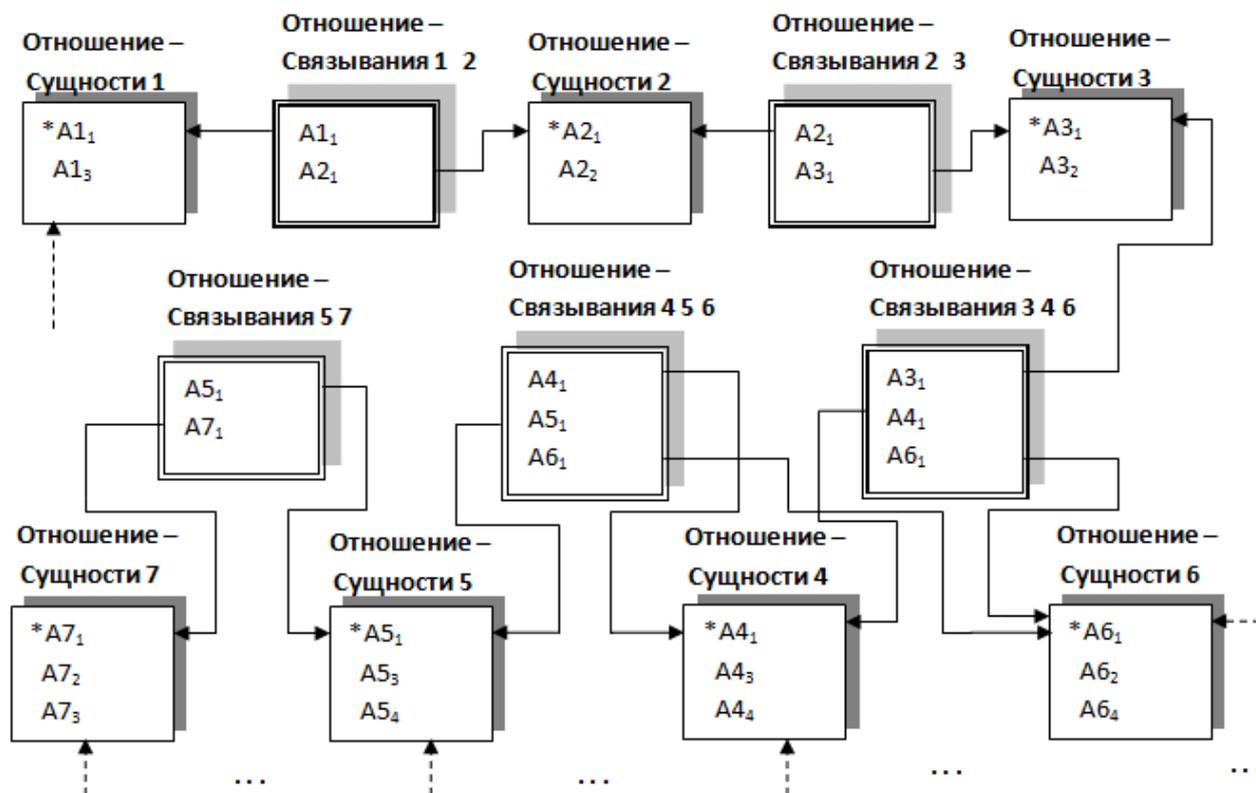


Рисунок 2 – Примерная логическая модель ОС2

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

- Простота методики, где нет необходимости исследовать на наличие разного рода зависимостей (транзитивной, многозначной, зависимости соединения);
- Конкретные шаги построения модели;
- Все отношения приведены к ДКНФ;
- Любую модель «сущность-связь» можно преобразовать в модель ОС2.

## ЛИТЕРАТУРА

1. Fagin R. A Normal Form for Relational Databases That Is Based On Domains and Keys // *CACM Transactions on Databases Systems*, 1981.