

## К ВОПРОСУ О МОДИФИЦИРУЕМОСТИ И БЕЗНОМАЛЬНОСТИ СХЕМЫ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ

Б.Е. Панченко

Институт кибернетики им. В.М. Глушкова НАН Украины,  
03187, проспект Академика Глушкова, 40,  
Тел.: (044) 5263603, [pr-bob@ukr.net](mailto:pr-bob@ukr.net)

Предложен новый подход к синтезу ДКНФ для произвольной предметной области. Дан новый критерий принадлежности схемы БД к ДКНФ. Показана единственность подхода к проектированию одновременно и модифицируемой, и безаномальной схемы реляционной БД, основанная на реляционном каркасе. Проведены численные исследования схемы конкретной предметной области. Обнаружен эффект ускорения работы БД за счет экономии на операциях соединений. Делается вывод о возможности применения данного подхода к проектированию схем информационных хранилищ.

A new approach to the DK/NF synthesis for an arbitrary domain is offered. A new criterion for the database scheme's appartenance to DK/NF is given. The uniqueness of the design approach of both modifiable and non-abnormal schemes of the relational BD, based on the relational framework, is shown. Numerous investigations of specific abstract domain's scheme have been carried out. The effect of acceleration of the database by saving on the compound operations is detected. The conclusion about the possibility of applying this approach to the design of information warehouse schemes is made.

### Введение

Наиболее важной частью любой информационной системы является ее база данных (БД). Синтез схемы БД – это сложный многокритериальный процесс, основанный на анализе предметной области (ПрО) [1]. Основными критериями оценки модели БД являются «целостность данных», «отсутствие аномалий» и «отсутствие избыточности данных» [2]. Однако со временем любой программный продукт подвержен изменениям. И схему БД, не отвечающую новым требованиям, необходимо подвергать обновлениям. Как правило, изменение схемы БД приводит к серьезным затруднениям и существенно повышает стоимость сопровождения. Следовательно, частое перепроектирование схемы БД приводит к реинжинирингу всей СУБД, что нежелательно для пользователей ввиду больших временных и финансовых затрат.

Указанные проблемы процесса проектирования зачастую возникают из-за низкой модифицируемости схемы БД [3 – 5]. Таким образом, именно модифицируемость является одним из самых важных критериев качества внедренного приложения. Однако это свойство невозможно реализовать на низко-нормализованных схемах, так как, с одной стороны, низкая нормализация является результатом плохой осведомленности проектировщика о всех причинно-следственных связях в ПрО. А с другой – низко-нормализованная БД подвергается значительным рискам. Существующая высокая вероятность аномалий вставки и удаления, продиктованная низким уровнем нормализации, снижает эксплуатационные характеристики и оперативных БД, и хранилищ данных. Но при этом, постоянные модификации приложений вносят в низко-нормализованную схему дополнительные источники ошибок. Все это значительно повышает стоимость сопровождения приложений.

В работе [6] обоснован алгоритм синтеза схемы БД в рамках реляционной модели данных (РМД) так, что одновременно удовлетворяются и критерии безаномальности и высокой модифицируемости.

### Доменно-ключевая схема БД

По аналогии с [7] введем понятие ограничения схемы отношения. *Z-ограничением* (или просто *ограничением*, если подразумевается совокупность  $Z$  отношений) будем называть отображение набора всех  $S$ -отношений на домен  $\{true, false\}$ . Такие ограничения на отношения вводятся, как правило, с помощью высказываний, написанных на заданном языке – таком, как исчисление предикатов 1-го порядка. На эти ограничения могут быть наложены еще и дополнительные ограничения.

Будем говорить, что отношение  $R$  *удовлетворяет* ограничению  $g$ , или, что  $g$  *содержится* в  $R$ , если после этого отображения, отношение  $R$  принимает значение *true*. Иначе, мы говорим, что  $g$  *ложно* (*false*) в  $R$ . К ограничениям также относят традиционные зависимости – ФЗ, МЗ, ЗПС, а также введенные в [7] зависимости домена – DD и зависимости ключа – KD.

Пусть  $X$  – набор атрибутов,  $g$  – единственное ограничение и  $G$  – набор ограничений (или единственное ограничение). Тогда очевидно, что  $G$  *логически включает*  $g$  (в контексте  $X$ ), или, что  $g$  – *логическое следствие*  $G$  (в контексте  $X$ ). Строго это означает, что всякий раз, когда  $G$  содержится в отношении с атрибутами  $X$ , тогда и  $g$  является таковым. Будем обозначать этот факт как обычное следствие  $G \Rightarrow g$ , если контекст  $X$  подразумевается. Например,  $\{A \rightarrow B, B \rightarrow C\} \Rightarrow A \rightarrow C$ .

Как и в [7], будем также обозначать зависимость домена  $IN(A, S)$ , где  $A$  – атрибут и  $S$  – множество. Означает, что для каждого кортежа значение атрибута  $A$  должно принадлежать  $S$ .

Зависимость ключа  $KEY(K)$ , где  $K$  – множество атрибутов, означает, что  $K$  является *ключом* (*key*), т.е., что в отношении нет двух кортежей, для которых значения  $K$  одинаковы.

С понятием ограничения реляционной схемы связано и понятие *выполнимости* реляционной схемы [7]. Схема выполнима, если для нее найдется, по крайней мере, один допустимый экземпляр схемы, и набор ее ограничений непротиворечив.

Теперь, по аналогии с [7], введем определение ДКНФ.

**Определение ДКНФ** [7]. Пусть  $R$  – реляционная схема, находящаяся в 1НФ. И пусть  $G$  – набор зависимостей  $DD(R) \cup KD(R)$  схемы.  $R$  находится в доменно-ключевой нормальной форме (ДКНФ), если  $G \Rightarrow g$  для любого ограничения  $g$  схемы  $R$ .

Выпишем из [7] основную теорему о ДКНФ. Она будет базовой для дальнейших выкладок.

**Теорема Фейджина о ДКНФ.** Выполнимая 1НФ-реляционная схема находится в ДКНФ тогда и только тогда, когда в ней нет аномалий вставки или удаления.

С учетом этого утверждения, доказанного в [7], рассмотрим *особое* отношение со схемой  $R(X,A)$ , названное так в [3]. При этом, по аналогии с [3], назовем также *особыми* следующие ограничения этой схемы:

$$DD(R) = IN(A, \forall A (A \neq \text{const})) \cap IN(X, \forall X (X \neq \text{const})),$$

$$KD(R) = KEY(X) \Rightarrow X \rightarrow A$$

Таким образом, особые ограничения – это такая совокупность, в которой ограничения на домены: любые значения атрибутов  $X$  и  $A$ , не являющиеся константой (и, в том числе, пустым множеством), ограничения на ключи:  $X$  – единственный ключ отношения, а дополнительные ограничения: наличие ФЗ  $X \rightarrow A$  – это естественное следствие из ограничения ключа.

Выпишем из [6] основные утверждения о безаномальности отношений, полученных на реляционном каркасе. Доказательства этих утверждений, приведенные в [6], выписывать не будем.

**Лемма 1.** Выполнимое отношение со схемой  $R_i(X_1, X_2, \dots, X_i, A_1, A_2, \dots, A_j)$  и ограничениями:  $(X_1 + X_2 + \dots + X_i)$  – единственный ключ отношения и ограничения на домены строго соответствуют ограничениям на домены в особых ограничениях, может быть заменено отношением с особой схемой  $R_i^*(X, A)$ , где  $X = (X_1 + X_2 + \dots + X_i)$  и  $A = (A_1 + A_2 + \dots + A_j)$  тогда, и только тогда, когда любая совокупность ФЗ исходной схемы является следствием ключа.

**Лемма 2.** В выполнимом каркасном отношении со схемой  $R_i(X_1, X_2, \dots, X_i)$ , полученном декартовым произведением атрибутов  $X_1, X_2, \dots, X_i$  (оператором роста [3]), с ограничениями:  $(X_1 + X_2 + \dots + X_i)$  – единственный ключ отношения и ограничения на домены соответствуют ограничениям на домены в особых ограничениях, любая совокупность ФЗ является следствием ключа.

Это значит, что в схеме шунтированного отношения  $R_i(X_1, X_2, \dots, X_i, A_1, A_2, \dots, A_j)$  любая совокупность ФЗ на частях единственного ключа как на детерминантах, причем так, что в этих зависимостях не участвует ни один из неключевых атрибутов  $(A_1, A_2, \dots, A_j)$ , не противоречит ограничениям ключа. Это очень важно для проектировщика, так как означает, что такая замена будет адекватной для разных совокупностей ФЗ, которыми обладают составные ключи при шунтировании обобщений декартовой зависимости (строго обоснованном в [6] частном случае многозначной зависимости и зависимости проекции-соединения, ДЗ  $\rightarrow$  МЗ  $\rightarrow$  ЗПС). Это утверждение обосновывает адекватность моделирования многоарной связи (с атрибутами) между множеством атомарных сущностей отношением со схемой, аналогичной схеме отношения, моделирующего единственную атомарную сущность. И полностью подтверждает идею Чена [9] о равносильности сущностей и связей.

**Лемма 3.** Выполнимая реляционная схема  $R(X,A)$  не имеет аномалий вставки или удаления тогда и только тогда, когда в ней отсутствуют ограничения, которые не следуют из особых ограничений.

Необходимо заметить, что описанное леммой 3 «минимальное» отношение также было рассмотрено Р. Фейджиным (в работе [7] – пример 3.10). Однако дополнительные ограничения на такую схему, названные там *зависимостями вложения* (когда каждое вхождение в  $X$  должно также входить и в  $A$ , обозначается  $X \subseteq A$ ), привели схему к аномалиям удаления. При этом в [7] указано, что более общая схема, в которой «единственными ограничениями являются ФЗ, может всегда простым способом преобразовываться в ДКНФ-схему БД...». Для каждой ФЗ  $X \rightarrow Y$  исходной схемы, мы формируем новую схему отношения с атрибутами  $XU$  и с единственным ограничением  $KEY(X)$ . Поэтому, по аналогии с [7], приведенные в [6] доказательства будем считать исчерпывающим всюду до тех пор, пока нет «контр-примера» отношения  $R(X,A)$  такого, что особые ограничения удовлетворяют  $R$ , но и такого, что найдутся такие иные ограничения-следствия особых  $D'(X,A)$  и  $K'(X,A)$ , что  $R$  станет аномальным.

Ведем определение схемы БД. Согласно [7] *схема базы данных* – это набор реляционных схем, каждая из которых представляет собой набор атрибутов и набор ограничений.

Каркасную совокупность отношений будем называть *замкнутой* и *связной*, если в ней нет ни одного отношения, у которого хотя бы одна часть простого или составного ключевого атрибута встречается в совокупности единственный раз. Интуитивно понятно, что такая совокупность моделирует замкнутые связные ПрО.

Доказанная в [6] лемма 3, а также доказанная в [3] теорема о шунтировании ДЗ позволяют сформулировать итоговые утверждения, завершающее построение базовой части теории реляционного каркаса.

Имеет место простое, но очень важное для проектирования безаномальных схем БД утверждение, доказанное в [6].

**Теорема 1.** Схема БД, построенная на совокупности шунтированных каркасных отношений, не имеет аномалий вставки и удаления, если и только если ограничения каждого отношения является логическим следствием особых ограничений.

Из этого утверждения следует простой вывод для проектировщика: если для некоторой ПрО схема БД проектируется на каркасной совокупности шунтированных отношений, и если для некоторых отношений из этой совокупности обнаруживаются ограничения на домены и ключи, противоречащие особым ограничениям, то эти ограничения из схем отношений должны быть исключены. В противном случае такая схема БД будет иметь аномалии.

Однако из теорем о полноте, единственности и замкнутости реляционного каркаса следует еще одно, даже более сильное утверждение.

**Теорема 2** (достаточное условие безаномальности реляционной схемы БД). Схема базы данных не имеет аномалий вставки и удаления, если она построена на замкнутой и связной совокупности шунтированных каркасных отношений, каждое из которых имеет ограничения, логически следующие из особых.

При этом совокупность отношений, каждое из которых находится в доменно-ключевой нормальной форме, будем называть ДКНФ-схемой БД. Проектировщику вполне достаточно знать, что, по крайней мере, на реляционном каркасе он может получить безаномальную ДКНФ-схему БД при условии, что все ограничения ПрО будут являться следствием особых.

Отметим, что выводы теорем 1 и 2 полностью соответствует идее Д.М. Кренке [2] о «единственности темы отношения». Если отношение моделирует высказывание с единственной темой, оно соответствует критериям ДКНФ. Единственный унарный или составной ключевой атрибут в этом контексте является семантически атомарным или составным многоместным предикатом [3, 8] такого высказывания. То есть – его «темой». Потому, что каждая шунтированная «ячейка» каркаса, то есть каждое актуальное отношение-связь, является высказыванием. А актуальность отношения-связи констатирует некий факт из ПрО.

По нашему мнению, именно Д.М. Кренке впервые ввел понятие «тема» как логическое следствие ключа: один ключ – одна тема. Он тем самым интуитивно подсказал, что у ключа отношения, если он единственен, появляется еще одна важная миссия, нежели только сохранность целостности отношения и доступ к данным, – быть носителем семантики отношения.

И вывод о том, что в отношении с ДКНФ-схемой не может быть более одного ключа и более одного детерминанта неключевых атрибутов, также предложил Д.М. Кренке. «Если в отношении имеется три атрибута  $R(A,B,C)$  и при этом  $(A,B) \rightarrow C$ , то не должно быть еще и зависимости  $A \rightarrow B$ . В противном случае по этой зависимости отношении должно быть декомпозировано» [2]. Отношение, построенное на реляционном каркасе, автоматически обладает единственной ФЗ.

### Актуализация ячеек каркаса

Наша цель – исследовать реляционный каркас, предложенный и формализованный в [3, 6, 8], а также продемонстрировать в его свойствах важные конкурентные преимущества, которые позволят проектировщику использовать эту совокупность реляционных отношений как базовый шаблон для проектирования схем БД произвольных ПрО. И при этом показать диаметрально противоположный к общепринятому подход в проектировании схем БД – не подстраиваться под каждую, часто необоснованную, специфику ограничений ПрО, а наоборот, на первом этапе отобрать в ПрО те ограничения, специфика которых является логическим следствием заведомо корректных. А ограничения ПрО, которые противоречат безаномальной схеме, реализовать на втором этапе в отдельных отношениях. Но и их схемы проектировать также на каркасе. Опыт проектирования схем БД позволяет заключить, что на этом шаблоне может быть реализована подавляющее большинство ограничений ПрО.

В работе [6] показано, что теория реляционного каркаса – это обоснование алгоритма синтеза ДКНФ-схемы БД [7]. А благодаря теореме об «устойчивости реляционного каркаса к модификациям» [8], еще и строгое обоснование гипотезы о максимальной степени модифицируемости ДКНФ-схемы БД, смело высказанной в [5]. Однако, с той лишь оговоркой, что [5] является частным случаем [3, 4]. При этом, существенным в практике каркасного анализа ПрО является то обстоятельство, что реляционный каркас показывает совокупность всех актуальных отношений. Поэтому статическая ER-диаграмма [9] становится не нужной.

В работе [2] Д.М. Кренке, цитируя Р. Фейджина, утверждает, что «поскольку отношения в ДКНФ не должны иметь аномалий модификации, предотвратить возникновение этих аномалий может сама СУБД, контролируя соблюдение ограничений доменов и ключей». В каркасной схеме БД ограничения от ПрО естественным образом могут быть наложены на домены и ключи атомарных в смысле [3] сущностей-объектов. Остальные отношения, которые моделируют произвольные связи атомарных сущностей-объектов, обладают строгой обусловленностью доменами и ключами унарных отношений. Если бы это было не так, каркасная схема не соответствовала бы критериям целостности. И не моделировала бы ПрО – целостной полной замкнутой и связной системы [10 – 12].

Замкнутую и связную каркасную совокупность шунтированных [3] отношений со схемами  $R_{i-1}(X_1, \dots, X_{i-1}, A_{i-1})$ ,  $R_i(X_1, \dots, X_i, A_i)$ ,  $R_{i+1}(X_1, \dots, X_{i+1}, A_{i+1})$  будем называть *обусловленной (или взаимно обусловленной)* доменами и ключами этой совокупности, если ограничения на домены и ключи не противоречат особым ограничениям, а также если выполняется ссылочная целостность совокупности – строгое тождество значений соответствующих одноименных ключевых атрибутов.

Ссылочная целостность подразумевает еще и ограничение на удаление кортежей, на которые ссылаются иные кортежи (хотя бы на один ключевой атрибут), пока они не удалены. Или разрешено каскадное удаление. Но это ограничение обеспечивает большинство СУБД автоматически.

В такой совокупности каждое отношение может выступать как в роли «поставщика» домена для ключевых атрибутов любой совокупности иных отношений, так и в роли «получателя» домена для своих ключевых атрибутов – ключевые атрибуты отношения следующего каскада каркасной совокупности обусловлены всеми предыдущими отношениями, то есть, доменами этих ключей. А домены следующих каскадов отношений обусловлены ключами предыдущих каскадов. При этом шунтирующие атрибуты также строго определяются соответствующими доменами из ПрО.

Выпишем из [10 – 12] основные определения простейших свойств системы, посредством которых построим начальную формализацию ПрО.

Каждая из систем  $P$  состоит из  $N$  элементов или разбивается на эти элементы. Известно, что элементы, из которых состоит система, не изолированы друг от друга, а каким-то образом взаимодействуют, связаны между собой, находятся в определенных реальных и/или логических отношениях.

Если элемент  $A_1$  связан с элементом  $A_2$ , говорят об одном отношении между двумя элементами. Если элемент  $A_1$  связан еще и с элементом  $A_3$ , то фиксируется второе отношение. Аналогично, если элементы  $A_2$  и  $A_3$  также связаны между собой, то имеется третье отношение и т. д. Обозначим общее число связей в системе  $P$  через  $s$ . Будем говорить, что  $P$  состоит из  $N$  элементов и  $s$  отношений.

Система  $P$  является *полной*, т.е. включает в себя все значимые элементы. Формально система из  $N$  элементов называется *полной*, если любой ее элемент может быть выражен в виде формулы через любые другие элементы этой системы.

Кроме того, система  $P$  является *замкнутой*. Условие замкнутости формулируется так: в системе  $P$  действует  $s$  и *только*  $s$  значимых для нее отношений.

Очевидно, что свойства полноты и замкнутости взаимно дополняют друг друга: то, чем служит полнота применительно к поэлементному составу системы, выступает как замкнутость применительно к ее отношениям.

Система  $P$ , наряду с полнотой и замкнутостью, подчиняется еще и ограничению *связности*, подразумевая под этим, что в системе нет изолированных или полуизолированных элементов. Каждый элемент связан со всеми другими посредством заданных в системе отношений.

Это приходится специально оговаривать, потому что, вообще говоря, возможна и отличная ситуация. Лишь часть элементов системы могла бы взаимодействовать между собой, тогда как другая никак не связана с первой.

В таком случае две части оказались бы изолированными друг от друга, и система  $P$  распалась бы на две самостоятельные подсистемы. Подобная картина противоречит "целостности" системы, т.е. произвольной ПрО.

Под связностью понимается даже нечто большее. Допустимо говорить о "сквозной", или "тотальной", связности - "все связаны со всеми". В дальнейшем, именуя систему  $P$  *целостной*, будем считать, что она обладает тремя перечисленными свойствами: полноты, замкнутости, связности.

В каркасной модели на систему наложено еще одно ограничение, называемое *простотой*. Наиболее четким определением *сложных* систем является следующее [11]: *сложной системой* называется система, в модели которой недостаточно информации для эффективного управления ею. В этом контексте каркасная модель данных оперирует с данными от *простых* систем – информации достаточно для эффективного управления системой. А что касается объема системы, а также произвола ее структуры, эти факторы не являются значимыми как для РМД в целом, так и для каркасной модели, как ее частного случая.

Элементы системы произвольно связаны между собой попарно, тройками, группами по четыре и т. д. Обозначим через  $k$  арность отношений: если отношения бинарные, то  $k=2$ , если тернарные, то  $k=3$  и т. д. В результате, чтобы определить количество отношений  $s$ , нужно пересчитать все возможные группы, состоящие из  $k$  элементов. Тогда  $s = C_N^k$ , где  $C_N^k$  – число сочетаний из  $N$  элементов по  $k$ .

В итоге используем классическую формулу числа сочетаний  $C_N^k = N! / (k!(N-k)!)$ , в котором величина  $k$  выступает в качестве параметра.  $C_N^k$  суммарно составляет  $2^N - 1$ , если  $k=1, N$ .

Описанная модель системы является множеством всех подмножеств, т.е. булеаном связей. Именно булеан связей и был выбран в [3, 4] как модель ПрО для «реляционной» формализации ER-модели Чена [9].

Дадим определение понятию «актуальная связь в ПрО». При этом оговорим, что в рамках настоящей работы время как параметр динамики системы не рассматривается. Рассматриваются лишь статические срезы ПрО, и потому анализируются закономерности синтеза схем БД лишь для этих срезов.

Под *актуальной ячейкой реляционного каркаса* будем понимать многоарное отношение  $R(X_1, X_2, X_3, \dots, X_n, A_j)$ , если оно имеет единственный составной ключ  $(X_1 + X_2 + X_3 + \dots + X_n)$ , шунтировано [3] хотя бы одним функционально зависящим (ФЗ) от всего ключа атрибутом  $A_j$ , так что  $(X_1 + X_2 + X_3 + \dots + X_n) \rightarrow A_j$ , а также входит в обусловленную совокупность так, что эта совокупность моделирует полную целостную замкнутую связную простую систему.

Необходимо подчеркнуть, что условие непротиворечивости ограничений (обусловленность схем совокупности отношений) необходимы проектировщику, чтобы схемы БД были *выполнимыми* [7]. Там же показаны примеры того, как ограничения могут быть непротиворечивыми и схема выполнимой, но логическая взаимосвязь между ограничениями отсутствовать – тогда схема становится аномальной.

Именно совокупность актуальных ячеек каркаса, или, другими словами, актуальная каркасная совокупность отношений, лишена аномалий вставки и удаления кортежей для любой ПрО при условии, что никакие дополнительные ограничения из ПрО не противоречат *особым* ограничениям, введенным в [6].

Тогда имеет место следующая

**Теорема 3** (о модифицируемой ДКНФ-схеме БД). Для того, чтобы схема базы данных обладала одновременно свойствами и модифицируемости, и безаномальности, необходимо и достаточно, чтобы данная схема была сформирована на актуальных ячейках реляционного каркаса, и ограничения на каждое отношение являлось бы логическим следствием особых ограничений.

Справедливость этого утверждения следует из справедливости теорем о безаномальности реляционного каркаса, строго доказанных в [6] (теоремы 7 и 8). Действительно, достаточность обеспечена, так как по условиям теоремы ограничения актуальных ячеек каркаса не только обусловлены (т. е., не противоречивы), но еще и являются следствием особых.

А необходимость доказывается на основании свойств реляционного каркаса, строго доказанных [6]. Из того факта, что совокупность отношений сформирована на актуальных ячейках реляционного каркаса, следует, что данная совокупность обладает модифицируемостью реляционного каркаса. Но модифицируемость реляционной схемы БД, построенной на каркасе, является следствием его полноты, единственности и замкнутости [6]. Поэтому никакая иная совокупность реляционных отношений не имеет аналогичных свойств. □

Этот же вывод можно получить еще и на основании доказанных в [3, 6] теорем о шунтировании обобщений ДЗ, полноте, единственности и замкнутости реляционного каркаса. Действительно, если бы каркас не обладал возможностью исключать разновидности ДЗ (не был бы шунтируемым в смысле [3]), то имела бы место аномалия типа ДЗ (МЗ или ЗПС). Если бы каркас в рамках реляционной модели был бы не единственным и не полным [3], то имела бы альтернативная совокупность отношений, обладающая аналогичными свойствами. Но тогда это противоречило бы целостности реляционной алгебры. Однако каркас единственен. Если бы он не был бы полным, то не обладал бы свойством независимости от произвола ПрО. И не обладал бы инвариантностью для произвольной совокупности запросов к схемам. А из теоремы о замкнутости путей нормализации каркаса [3] следует, что на каркасе может быть построена вся совокупность схем. Пересечение всех перечисленных свойств обеспечивается единственно возможной схемой БД – 5НФ [7]. Соответствие шунтированного каркасного отношения для любых ПрО критериям 5НФ доказано строго [3].

Но теорема о замкнутости путей нормализации реляционного каркаса [3] позволяет обосновать еще и тот факт, что каркасная совокупность 5НФ-отношений - схема более сильная, потому, что каркас обеспечивает еще и полную взаимообусловленность всех атрибутов, чего, по сути, и требует определение ДКНФ [7]. В этой схеме присутствуют только ключевые атрибуты, обусловленные только друг другом и своими доменами. А также шунтирующие атрибуты, также обусловленные только ключевыми атрибутами, доменами ключевых атрибутов и своими доменами. Таким образом, для любых ограничений ПрО, наложенных на домены и ключи этой совокупности, каждое отношение из этой совокупности удовлетворяет критериям ДКНФ [7] только тогда, когда эти ограничения не противоречат особым.

## Результаты экспериментальных исследований

Взаимная обусловленность всех ключевых атрибутов ДКНФ приводит к потребности формирования дополнительных процедур отслеживания ссылочной целостности БД. Но поскольку такие процедуры легко поддаются формализации и унификации, эта ситуация не может быть отнесена к разряду «критических». С другой стороны, применяя единообразную процедуру индексирования ключевых полей всех связанных отношений, пользователь получает заметные преимущества в скорости доступа ко всем данным приложения.

В работе проведен специализированный эксперимент, основной характеристикой которого является время доступа к любому атрибуту любого отношения исследуемой БД. Как известно, самой ресурсоемкой операцией реляционной алгебры являются соединения отношений [1]. Эта операция применяется при выполнении запросов к БД более низкой «нормальности» – не выше 3НФ. Отличительной же особенностью именно ДКНФ является то, что в модели замкнутой системы [11, 12] все отношения уже соединены. И целостность БД гарантирует наличие любых ключевых данных для любых соединений. Поэтому отсутствие операции соединения приводит к значительной экономии времени доступа к БД. Если, конечно же, иное не навязано пользователю реализацией конкретной СУБД.

Для экспериментального исследования каркасного метода проектирования схемы БД и проведения указанного численного эксперимента была выбрана общеизвестна ПрО: «Приемная комиссия Университета».

Фрагмент итоговой диаграммы атомарных и составных сущностей-объектов, то есть их связей, приведен на рисунке 1. Опишем подробно схему каждой сущности-объекта. Для этого проведем анализ семантики фрагмента ПрО.

Сущность-объект *ВУЗ* – атомарная для данной ПрО - справочник *ВУЗов* донного университета. Подчеркивается «данной ПрО», потому приведен лишь фрагмент ПрО. Атомарность или слабость устанавливается именно в сравнении границ ПрО. В данном исследовании схема сущности-объекта *ВУЗ* упрощена. И ей назначен статус атомарной. Ее зависимость от сущности-объекта *ГОСУДАРСТВО*, а потому от ключевого атрибута *КодГосударства*, также можно пренебречь. Таким образом, *КодВУЗа* – это суррогатный унарный ключ сущности *ВУЗ*. Наименование конкретных *ВУЗов* хранится в атрибуте *Название*. Атрибут *СокрНаз* – это аббревиатура *ВУЗа*. Очевидно, что все иные атрибуты могут быть встроены аналогично.

Сущность-объект *ОТДЕЛЕНИЯ* – также атомарная сущность. Отражает информацию об отделениях. Как правило, существует два отделения – дневное и заочное. Иногда имеется еще и вечернее. Ключ – унарный суррогатный – *КодОтдел*.

Сущность-объект *СПЕЦИАЛЬНОСТЬ* – атомарная. Это – справочник специальностей в соответствии с государственным классификатором. Таблица, которая моделирует эту сущность-объект, хранит данные о названии специальности и иных ее свойствах. *КодСпец* – унарный суррогатный ключевой атрибут.

Сущность-объект *ДИСЦИПЛИНЫ* – слабая сущность-объект – общегосударственный справочник дисциплин, каждая из которых зависит от сущности-объекта *СПЕЦИАЛЬНОСТЬ*. *КодСпец+КодДисц* – суррогатный бинарный ключевой атрибут.

Сущность-объект *АБИТУРИЕНТ* – атомарная сущность-объект, которая не зависит от иных сущностей-объектов. Его условной зависимостью от атрибута *КодГосударства* можно также пренебречь. Поэтому в данной ПрО имеет унарный суррогатный ключевой атрибут – *КодАбитуриент*. А отношение, которое моделирует эту сущность-объект – это перечень всех абитуриентов Украины, которые подали заявления именно в этот университет. Т.е., в те или иные его ВУЗы. Содержит атрибуты *ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, ГОД РОЖДЕНИЯ, МЕСТО РОЖДЕНИЯ* и т.п.

Сущность-объект *ФАКУЛЬТЕТ* – слабая сущность-объект, которая обладает зависимостью существования. Однако может моделироваться и постсвязной составной (то есть связью) от независимых между собой атомарных сущностей-объектов *ВУЗы* и *ФАКУЛЬТЕТЫ В ГОСУДАРСТВЕ*. Тут это справочник факультетов *ВУЗов* университета. *КодВУЗа+КодФак* – суррогатный составной бинарный ключевой атрибут. *НАЗВАНИЕ* факультета – один из неключевых атрибутов сущности-объекта.

Сущность-объект *ПРЕПОДАВАТЕЛИ* – атомарная сущность-объект – перечень преподавателей. Зависимостью от сущностей-объектов *ОБЛАСТЬ, ГОРОД* или *СПЕЦИАЛЬНОСТЬ* в этой ПрО можно пренебречь. Поэтому ключ – унарный атрибут *КодПрепод*. Неключевые атрибуты – ФИО, степень, звание, дата рождения и т.п.

Составная сущность-объект *ПРЕПОДАВАТЕЛИ В ВУЗе* – это связь сущностей-объектов *ПРЕПОДАВАТЕЛИ* и *ВУЗы*. Поэтому ключ – бинарный составной атрибут *КодВУЗа+КодПрепод*. Атрибутами связи могут быть *ДАТА, ПРИМЕЧАНИЕ* и т. п.

Сущность-объект *СПЕЦИАЛЬНОСТИ В ВУЗах* – составная бинарная сущность-объект (связь). Ключ – бинарный составной атрибут – *КодВУЗа+КодСпец*. Моделирующее ее отношение отфильтровывает пакет специальностей из отношения *СПЕЦИАЛЬНОСТИ* именно для данного университета. Атрибут связи – текущее *НАЗВАНИЕ* специальности именно в этом *ВУЗе, ДАТА* ее открытия и т. п.

Сущность-объект *ДИСЦИПЛИНЫ В ВУЗах* – составная тернарная сущность-объект (связь) между сущностями-объектами *ДИСЦИПЛИНЫ, ВУЗы* и *СПЕЦИАЛЬНОСТИ*. Ключ – тернарный составной атрибут – *КодВУЗа+КодСпец+КодДисц*. Атрибуты связи – текущее *НАЗВАНИЕ* дисциплины, *ДАТА* открытия, суммарные плановые показатели и т.п.

Сущность-объект *АБИТУРИЕНТЫ ВУЗов* – составная сущность-объект (связь), которая связывает сущность-объект *АБИТУРИЕНТЫ* и сущности-объекты *ВУЗы* и *СПЕЦИАЛЬНОСТЬ*. Поэтому имеет тернарный ключевой атрибут – *КодВУЗа+КодСпец+КодАбитуриент*. Моделирующее отношение – это перечень абитуриентов, которые подали заявления именно в эти *ВУЗы* данного университета и именно на эти *СПЕЦИАЛЬНОСТИ*. Обладает атрибутами связи – *ДАТА* заявления, *ПРИМЕЧАНИЕ* и т.п.

Сущность-объект *УЧЕБНАЯ ПРОГРАММА* – является составной сущностью-объектом. В этой ПрО она является результатом связи семи сущностей-объектов – *ВУЗ, СПЕЦИАЛЬНОСТЬ, ОТДЕЛЕНИЕ, ФАКУЛЬТЕТ, ДИСЦИПЛИНА, ПРЕДМЕТ И СЕМЕСТР*. Составной септарный ключевой атрибут: *КодВУЗа+КодСпец+КодОтдел+КодФак+КодДисц+КодПредм+КодСеместр*. Атрибуты связи – *ЧИСЛО ЧАСОВ, ПРИМЕЧАНИЕ* и т.п. Причем сущность *УЧЕБНАЯ ПРОГРАММА* не зависит от сущностей-объектов *ПРЕПОДАВАТЕЛИ, ГРУППЫ СТУДЕНТОВ, АУДИТОРИИ, НЕДЕЛЯ* и *ДЕНЬ*. Если в состав ключевых атрибутов ввести еще и эти атрибуты, получим составную сущность-объект (связь) *РАСПИСАНИЕ ЗАНЯТИЙ*.

Сущность-объект *ВСТУПИТЕЛЬНЫЕ ЭКЗАМЕНЫ* – септарная составная сущность-объект (связь). Ключ – составной септарный атрибут: *КодВУЗа+КодСпец+КодФак+КодОтдел+КодПредм+КодПреп+КодАбитуриент*. А атрибуты связи – *ОЦЕНКА, ДАТА* и т.п.

Можно сравнить сущность-объект *ВСТУПИТЕЛЬНЫЕ ЭКЗАМЕНЫ* и сущность-объект *ЭКЗАМЕНЫ СЕССИИ*. Большая часть составного ключевого атрибута совпадает, так как совпадает происхождение смысла этих различных сущностей. Однако имеется и отличительные звенья. Вместо абитуриента, так сказать, субъектом связи выступает уже сущность-объект *СТУДЕНТ*, которая именно в этой ПрО имеет свой тернарный ключевой атрибут – *КодВУЗа+КодСпец+КодСтуд*. А атрибут *КодАбитуриент*, который был ключом в таблице *АБИТУРИЕНТ*, в этой таблице уже является неключевым справочным атрибутом. Именно эта часть совокупности отношений уже относится к приложению *УЧЕБНЫЙ ПРОЦЕСС*.

В этой ПрО появляется и слабая сущность-объект *ГРУППА*, которая объединяет совокупность студентов одной специальности и года зачисления в *ВУЗ*. Она, как правило, имеет тернарный ключ *КодВУЗа+КодСпец+КодГруппы*. А год зачисления, то есть год формирования группы, как правило, является неключевым атрибутом (на диаграмме не показан). Следовательно, в этой части приложения обязательно моделируется и сущность-объект *СТУДЕНТЫ В ГРУППАХ*, то есть составная сущность-объект (связь) между

сущностью-объектом *СТУДЕНТЫ* и сущностью-объектом *ГРУППЫ*. И, в свою очередь, ее ключ - квартарный составной атрибут *КодВУЗа+КодСпец+КодГруппы+КодСмуд*.

Однако, как правило, атрибут *КодСмуд* не зависит от атрибута *КодГруппы*, так как группы – это условное разделение множества студентов, которое упрощает управление ими. Поэтому группы имеют в своем составе небольшое число людей – элементов множества. Именно небольшой диапазон изменения значений этого атрибута и влияет на то, что пользователь ПрО назначает домен значений атрибута *КодСмуд* так, что одновременно кодирует в его структуре и группу. Поэтому эта связь является лишь справочной. А значит в данной ПрО в сущности-объекта *ЭКЗАМЕНЫ СЕССИИ* атрибут *КодГруппы* участия не принимает.

В качестве ключевых для сущности-объекта *ЭКЗАМЕНЫ СЕССИИ* добавляются еще и атрибуты, которые обуславливают специфику сессии – *НОМЕР ПОПЫТКИ* (первая, вторая или даже четвертая), а также *ТИП СЕССИИ* – зимняя или летняя.

Таким образом, сущность-объект *ЭКЗАМЕНЫ СЕССИИ* – составная нонарная сущность-объект (связь).  
 Ключ – составной нонарный атрибут:  
*КодВУЗа+КодСпец+КодФак+КодОтдел+КодПредм+КодПреп+КодСмуд+НомПопыт+КодСессии*.  
 А атрибуты связи – оценка, дата и т.п.

На рис. 1 показана не традиционная ER-диаграмма Чена [9]. Здесь дуги играют роль не связей между сущностями-объектами, а указателей на основные взаимообусловленности доменов и ключей. А значит и маршруты отслеживания целостности для СУБД. Причем, очевидно, что соответствие имен ключевых атрибутов и означает их взаимность. Впервые аналогичные диаграммы были предложены в [13].

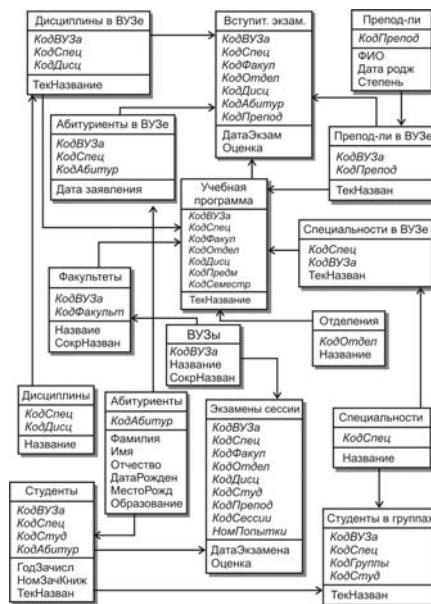


Рис. 1. ДКНФ-схема БД фрагмента ПрО «Университета»

На рис. 2 показан график, иллюстрирующий рост времени ( $t$  по оси ординат, в секундах) доступа к данным при получении одной записи из увеличивающихся групп кортежей ( $z$  по оси абсцисс, в миллионах записей) при запросе на соединение 4 отношений, находящихся в ЗНФ (кривая 2). И значительную общую экономию времени на индексную выборку этой же записи из одного квартарного отношения в ДКНФ (кривая 1), а также слабый прирост времени при увеличении числа записей. При этом результирующее отношение обладало в среднем от 100 до 200 кортежей.

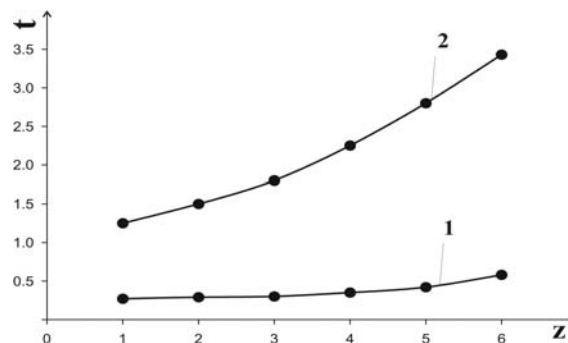


Рис. 2. ПрО «Университет», 4 отношения

## **Вывод**

Полученные результаты уточняют выводы многих авторов (например, серия работ авторов [14]), что высокую эффективность работы БД (скоростные показатели) можно получить лишь путем денормализации. Даже если в некоторых локальных ситуациях эффективность денормализованных схем выше, этот результат не может претендовать на универсальный подход, потому что он также плохо поддается алгоритмизации и типизации, как и сама нормализация. Эффективность же ДКНФ является следствием строгой обоснованности и прогнозируемости.

Благодаря реляционному каркасу [3, 4, 6, 8] получена возможность для произвольной Про разработать алгоритм автоматизированной декомпозиции и синтеза схемы реляционной БД в форме, относящейся к максимально высокой степени нормальности. При этом каркасная схема БД получает дополнительные конкурентные преимущества, важные для проектировщиков-практиков.

1. *Мейер Д.* Теория реляционных баз данных. – М.: 1987. – 608 с.
2. *Кренке Д.М.* Теория и практика построения баз данных. – СПб.: Питер, 2003. – 800 с.
3. *Карпуша В.Д., Панченко Б.Е.* Моделирование и проектирование реляционных баз данных. Учебное пособие. – Суми: Изд. СумДУ, 2010. – 385 с.
4. *Пат. 92248.* Способ обобщенного размещения данных с учетом модифицируемости структуры хранилища /Панченко Б.Е.// Промислова власність. – 2010. – № 19. – С. 3.131-3.132.
5. *Алтайбек А.А.* Проектирование баз данных на основе доменно-ключевой нормальной формы // Вестник КазНУ. – Алматы, 2009. № 1.– С. 111 – 118.
6. *Панченко Б.Е.* Каркасное проектирование доменно-ключевой схемы реляционной базы данных // Кибернетика и системный анализ. – 2012. – № 3. – С. 174 – 187.
7. *Fagin R.* A Normal Form for Relational Databases That Is Based on Domains and Keys// ACM Transactions on Database Systems, – 1981, – V. 6, N 3. – P. 387 – 415.
8. *Панченко Б.Е., Писанко И.Н.,* Свойства реляционного каркаса на множестве семантически атомарных предикатов // Кибернетика и системный анализ. – 2009. – № 6. – С. 120 – 129.
9. *Chen P.P.* The Entity-Relationship Model: toward a unified view of data // ACM Trans. on Data base systems. – 1976. – V. 1, N 1. – P. 9 – 36.
10. *Степанов А.И.* Число и культура. – М., 2004 – 832 с.
11. *Волкова В.Н., Денисов А.А.* Основы теории систем и системного анализа. Учебник. – СПб.: Изд. СПбГТУ, 2001 – 512 с.
12. *Перевозчикова О.Л.* Основы системного анализа объектов и процессов компьютеризации. Учебник. – Киев: Изд. дом “КМ Академия”, 2003. – 247 с.
13. *Панченко Б.Е., Гайдабрус В.Н., Церковицкий С.Л.* Создание сетевых информационных комплексов // Компьютеры плюс программы. – 1993. – № 5 (6). – С. 46 – 50.
14. *Кунгурцев А.Б., Зиноватная С.Л.* Модель реструктуризации реляционной базы данных путем денормализации схемы отношений // Тр. Одес. политехн. ун-та. – Одесса: ОНПУ, 2006. – № 2(26). – С. 105 – 111.