На правах рукописи

РЕДРЕЕВ Павел Григорьевич

РАЗРАБОТКА И ИССЛЕДОВАНИЕ ОБОБЩЕННОЙ ТАБЛИЧНОЙ МОДЕЛИ ДАННЫХ СО СПИСОЧНЫМИ КОМПОНЕНТАМИ

05.13.17 – теоретические основы информатики

Автореферат
диссертации на соискание ученой степени
кандидата физико-математических наук

Челябинск – 2011

Работа выполнена в Омском филиале Учреждения Российской академии наук Института математики им. С.Л. Соболева Сибирского отделения РАН

Научный руководитель: доктор технических наук, профессор
ЗЫКИН Сергей Владимирович

Официальные оппоненты: доктор технических наук, профессор
МОКЕЕВ Владимир Викторович

кандидат физико-математических наук
ТУРДАКОВ Денис Юрьевич

Ведущая организация: Учреждение Российской академии наук
Институт вычислительной математики
и математической геофизики Сибирского
отделения РАН

Защита состоится 28 сентября 2011 г. в 12 часов на заседании диссертационного совета Д 212.298.18 при Южно-Уральском государственном университете по адресу: 454080, г. Челябинск, пр. Ленина, 76, ауд. 1001.

С диссертацией можно ознакомиться в библиотеке Южно-Уральского государственного университета.

Автореферат разослан 25 августа 2011 г.

Ученый секретарь
диссертационного совета М.Л. Цымблер

Общая характеристика работы

Актуальность работы. Актуальной проблемой для многих предприятий является оперативная обработка и анализ накопленной информации. Имея оперативный доступ к огромным массивам данных, сотрудники предприятия не в состоянии сделать из них какие-либо выводы без использования специальных методов представления и обработки информации. Наиболее популярным способом решения указанной проблемы в настоящее время является технология оперативной аналитической обработки данных OLAP (online analytical processing). Основой OLAP-технологии является построение гиперкубического (многомерного) представления данных.

Не менее актуальна проблема автоматизации анализа данных и для пользователей сравнительно небольших баз данных, поскольку одни и те же данные приходится многократно реорганизовывать вручную для поиска скрытых в них закономерностей.

Многие аналитики OLAP придерживаются точки зрения, что кубическое представление данных должно быть постоянно хранимым и периодически обновляемым из операционной базы данных (MOLAP). Основным аргументом в пользу такого дублирования данных выдвигается требование минимального времени отклика системы на запросы пользователя. При этом предполагается, что на одном гиперкубе будут удовлетворены все потребности пользователя в анализе данных. Другой подход заключается в преобразовании схемы исходной операционной базы данных в "звезду" или "снежинку" (ROLAP). Такой подход нарушает принцип независимости данных, в частности независимость схемы операционной базы данных от места и способа использования данных. Общий недостаток этих двух подходов в регламентированности предполагаемых операций анализа данных. И если пользователю потребуется по-иному сгруппировать данные, то ему придется не один рабочий день потратить на реорганизацию данных совместно со специалистом по информационным технологиям. Это и является основным сдерживающим фактором широкого распространения технологий аналитической обработки данных.

В данной работе предлагается следующая технологическая последовательность обработки данных:

1. Исходные данные должны быть представлены в реляционном нормализованном виде, и к ним обеспечивается доступ по технологии OLTP (online transaction processing).

2. Пользовательское представление данных в виде *композиционной таблицы*, реализующее технологию OLAP, обеспечивается инструментарием, преобразующим исходные данные в необходимое представление данных.

3. Представление данных в виде *композиционной таблицы* далее используется для визуального, статистического и т.п. анализа данных.

Существенные затраты времени для формирования схемы и реализации гиперкуба в данной работе предлагается сократить за счет автоматизации этого

процесса с использованием свойств схемы исходной операционной базы данных.

Цель и задачи исследования. *Цель* данной работы состояла в автоматизации формирования схемы и представления данных в виде *композиционной таблицы* со списочными компонентами из исходного реляционного представления данных. При этом должны быть реализованы логические и контекстные ограничения на исходные данные. Для достижения этой цели необходимо было решить следующие задачи:

1. Разработать модель многомерного представления данных на плоскости – *композиционную таблицу*.
2. Исследовать свойства *композиционной таблицы*, в том числе способы автоматического формирования иерархий в измерениях.
3. Исследовать свойства промежуточного представления данных – *таблицы связанных соединений*.
4. Разработать и реализовать алгоритмы формирования *таблицы связанных соединений*, *композиционной таблицы* и иерархий в измерениях.
5. Провести вычислительный эксперимент для построения диагностической шкалы на исходных данных пациентов кардиологического диспансера.

Методы исследования. При выполнении работы были использованы методы межмодельных коммутативных преобразований, теория проектирования реляционных баз данных, методы анализа данных.

Научная новизна работы заключается в следующем:

1. Разработаны модель и алгоритмы формирования *композиционной таблицы*.
2. Исследованы свойства и условия существования промежуточного представления данных – *таблицы связанных соединений*.
3. Разработан алгоритм автоматизированного формирования иерархий в измерениях.
4. Разработан алгоритм автоматического формирования *контекстов измерений* и *контекста приложения* и доказана корректность построения представления *композиционной таблицы*.
5. Реализовано программное обеспечение, формирующее представление *композиционной таблицы*, и на его основе разработана диагностическая шкала оценки тяжести артериальной гипертензии.

Теоретическая ценность работы. Разработана теория и алгоритмы формирования представления *композиционной таблицы* на основе теории межмодельных преобразований данных.

Практическая ценность работы. Реализовано программное обеспечение формирования *композиционной таблицы* на основе межмодельных преобразований данных при наложении ограничений на данные. С использованием программного обеспечения рассчитана шкала для диагностирования пациентов кардиологического диспансера.

Результаты диссертационной работы могут применяться при создании

OLAP-систем и в учебном процессе при подготовке бакалавров по направлению «Информатика и вычислительная техника». Разработанные методы, алгоритмы и программы могут быть использованы в научных исследованиях в области систем управления базами данных и аналитической обработки данных.

Достоверность научных результатов, полученных в диссертации, подтверждается строгими математическими доказательствами и экспериментальными исследованиями.

Апробация работы. Результаты работы доложены на следующих конференциях и семинарах:

- Седьмая международная конференция «Перспективы систем информатики». Рабочий семинар «Наукоемкое программное обеспечение». – Новосибирск, 2009.
- Всероссийская конференция с международным участием «Знания – Онтологии – Теории» (ЗОНТ-09). – Новосибирск, 2009.
- Школа-семинар «Новые алгебро-логические методы решения систем уравнений в алгебраических системах». – Омск, 2009.
- Семинар лаборатории МППИ ОФ ИМ СО РАН. – Омск, 2010.

Публикации. По теме диссертационной работы опубликовано 7 работ, из них статьи в изданиях из перечня ВАК – 3. Все публикации написаны без соавторов, кроме [3], в которой Зыкину С.В. и Чернышеву А.К. принадлежит постановка задачи, решение задачи принадлежит Редрееву П.Г. Получено 1 свидетельство об отраслевой регистрации разработки.

Структура и объем диссертации. Диссертационная работа состоит из введения, пяти глав, заключения и списка литературных источников, изложенных на 114 страницах, объем библиографии - 101 наименование.

Содержание работы

Во введении обоснована актуальность темы, сформулирована цель работы, представлены основные результаты диссертационной работы.

В первой главе «Подходы к реализации OLAP-технологии» анализируются виды систем OLAP: реляционные системы OLAP (ROLAP), многомерные системы OLAP (MOLAP), гибридные системы OLAP (HOLAP).

Концептуальные модели OLAP подразделяются на три основных класса: расширения реляционной модели, кубические модели и HOLAP – объединение технологий ROLAP и MOLAP. К первому классу относятся также разработки, предлагающие дополнение существующих языков запроса дополнительными конструкциями.

Для представления данных в OLAP-системах используются многомерные модели данных, являющиеся гиперкубами, то есть обобщением электронных таблиц на произвольное количество измерений (dimensions). В многомерных моделях данные рассматриваются либо как меры (measures), которые являются числовыми значениями, либо как текстовые измерения. Меры – это величины, подвергаемые анализу по измерениям. Измерение включает в себя уровни из-

мерения, позволяющие пользователю анализировать меры с различной степенью детализации. Из уровней измерения могут формироваться иерархии. Наличие иерархий позволяет осуществлять выполнение таких часто используемых для анализа данных операций как roll-up и drill-down. Конкретное значение уровня иерархии называется элементом (member).

В работах, посвященных многомерным моделям данных, в иерархиях измерений предыдущий уровень измерения функционально определяет последующий, в измерениях поддерживается небольшое количество различных видов иерархий. В работе Педерсена для рассматриваемой многомерной модели реализованы нерегулярные иерархии, возникающие в различных приложениях.

В данной работе предлагается автоматизированное формирование иерархий в измерениях. Кроме того, предполагается, что основой аналитической работы пользователя является формирование новых гиперкубов, а не многократное формирование реализации одного и того же гиперкуба. Следовательно, основное внимание необходимо акцентировать на сокращении времени формирования схемы нового гиперкуба, а формирование представления гиперкуба должно быть выполнено автоматически алгоритмами, соответствующими выбранному классу схем.

В качестве основы для автоматизации формирования представления гиперкуба предлагается использовать формальное определение промежуточной и целевой моделей данных, задающих не только схемы, но и способы формирования представлений.

Для создания инструментария формирования пользовательских приложений осуществляется разработка целевой модели данных и построение межмодельного отображения между целевой моделью данных и исходной моделью данных.

Рассмотренная в данной работе модель *композиционная таблица* является обобщением модели «семантическая трансформация» на случай списка значений в одной ячейке, разделенных знаками препинания.

Во второй главе «Формирование представлений данных для аналитической обработки» описываются принципы формирования гиперкубического представления данных, рассмотрены алгоритмы формирования таблицы соединений и гиперкубического представления данных. Рассмотрены возможные виды накладываемых ограничений на данные. Рассмотрен алгоритм автоматического формирования иерархии в измерении.

Для автоматизации построения *композиционной таблицы* предлагается следующая последовательность формирования ее представления:

1. Пользователь из списка атрибутов БД формирует множества атрибутов: измерения X, Y_1, Y_2,…, Y_N и меры Z_1, Z_2,…, Z_N. $X \cap Y_i = \varnothing$, $(X \cup Y_i) \cap Z_i = \varnothing$, $i=1,2,\ldots N$.

2. Автоматическое формирование иерархий измерений для множеств атрибутов X, Y_1, Y_2,…, Y_N.

3. Задаются логические ограничения на измерения $F_0(X)$, $F_1(Y_1)$, $F_2(Y_2)$,…, $F_N(Y_N)$. По умолчанию каждая формула есть конъюнкция условий определенности (***IS NOT NULL***) для атрибутов измерения.

4. Формирование *контекстов измерений* $C_0, C_1,\ldots, C_N$. (некоторые *контексты* могут быть пустыми, а некоторые – *псевдоконтекстами*).

5. Формирование *контекста приложения* C_{full} и соответствующей реализации *таблицы связанных соединений* c со схемой C и логическим ограничением $F(C)=F_0(X)\wedge(F_1(Y_1)\vee F_2(Y_2)\vee\ldots\vee F_N(Y_N))$

6. Формирование реализаций измерений $X, Y_1, Y_2,\ldots, Y_N$ с сортировкой значений в соответствии с иерархией.

7. Формирование реализации (представления) *композиционной таблицы* (заполнение значений мер на соответствующих местах таблицы).

Рассмотрим правило формирования логического ограничения $F(C)$.

Правило 2.1. Каждое выражение F_i, $i=0,1,\ldots,N$, должно быть представлено в виде дизъюнкции элементарных формул: $F_i = F_{i,1}\vee F_{i,2}\vee\ldots\vee F_{i,m(i)}$, где $m(i)$ – массив целых чисел. Каждая элементарная формула является конъюнкцией атомарных условий: $F_{i,j} = F^1_{i,j} \wedge F^2_{i,j} \wedge \ldots \wedge F^p_{i,j}$, где $F^s_{i,j}$ = $=A_q\Theta$<выражение>, <выражение> – константа либо атрибут A'_q, Θ -операция сравнения.

Далее рассмотрим правило вычисления выражения $F(C)$.

Правило 2.2. Пусть t – произвольный кортеж, определенный на множестве атрибутов V, если какой-либо терм $F^s_{i,j}$ формулы F не определен на множестве V (атрибуты A_q и/или A'_q не принадлежат множеству V), то терм $F^s_{i,j}$ заменяется значением ***TRUE*** независимо от операции Θ.

Для представления данных *композиционная таблица* множества атрибутов X и Y_j $(j=1, 2, \ldots, N)$ являются обобщенными координатами и могут рассматриваться как измерения.

В качестве уровней измерения будем использовать атрибуты исходной базы данных. Пусть L – множество атрибутов X или Y_j *композиционной таблицы.*

Определение 2.1. Схема иерархии – это связный ориентированный ациклический граф $H=(A, E)$, где A – множество атрибутов, E – множество дуг.

Определение 2.2. Пусть V, D – атрибуты. H – схема иерархии, тогда $V\prec D$, если в **H** существует путь из вершины V в D.

Для задания частичного порядка на множестве атрибутов, входящих в функциональные и многозначные зависимости, используется следующее эвристическое правило.

Правило 2.3. Атрибуты из множества атрибутов, принимающего меньшее количество значений, располагаются в иерархии выше, чем атрибуты из множества, принимающего большее количество значений.

Для функциональной зависимости $V\rightarrow D$, где V и D – множества атрибутов, атрибуты из D располагаются в иерархии выше, чем атрибуты из V.

Для многозначной зависимости $V \twoheadrightarrow D\ (E)$, где V, D, E – множества атрибутов, атрибуты из V располагаются в иерархии выше, чем атрибуты из $D\cup E$.

Некоторые последовательности уровней могут многократно использоваться в иерархиях измерений различных гиперкубов или задаваться в заголовках пользовательских представлений данных. Следовательно, для данных атрибутов пользователю необходимо предоставить возможность корректировки иерархий, сформированных автоматически алгоритмом.

Пусть $U = \{A_1, A_2, ..., A_n\}$ – некоторое множество атрибутов, R – исходное отношение, определенное на всем множестве U и удовлетворяющее зависимостям DEP, и $\{R_1, R_2, \ldots, R_k\}$ – множество отношений (декомпозиция R), определенных на подмножествах атрибутов множества U.

Определение 2.3. Декомпозиция $\{R_1, R_2, \ldots, R_k\}$ обладает свойством соединения без потерь информации (СБПИ), если для любой реализации отношения R, удовлетворяющей множеству зависимостей DEP, выполнено:

$$R = \pi_{R1}(R) \bowtie \pi_{R2}(R) \bowtie \ldots \bowtie \pi_{Rk}(R),$$

где $\bowtie$ – операция естественного соединения, $\pi_{Rj}(R)$ – проекция отношения R по атрибутам отношения R_j.

Пусть $C_x = \{R_1, R_2, ..., R_k\}$ – произвольное множество отношений реляционной БД.

Определение 2.4. Множество C_x будем называть *контекстом*, если оно удовлетворяет свойству СБПИ на зависимостях DEP.

Для повышения уровня автоматизации работы пользователя и снижения требований к его квалификации формирование *контекстов* осуществляется по исходным множествам атрибутов X, Y_i, Z_i $(i = 1, 2, \ldots, N)$.

Пусть $P_x = \{R_1, R_2, ..., R_k\}$ – произвольное множество отношений реляционной БД.

Определение 2.5. Множество P_x будем называть *псевдоконтекстом*, если для него не обязательно выполнение свойства СБПИ на зависимостях DEP.

Способ формирования *псевдоконтекста* аналогичен способу формирования *контекста*, за исключением дополнения отношений для удовлетворения свойства СБПИ.

В качестве дополнительной информации для направленного выбора отношений при формировании *контекстов* можно использовать зависимости включения. Зависимости включения реализуются в виде связей на схеме БД. При этом, от отношения R_i к R_j может быть установлена связь ***1:1*** либо ***1:M***, где R_i – главное отношение, R_j – подчиненное.

Для построения множества отношений, с наибольшей вероятностью удовлетворяющего свойству СБПИ используется следующее эвристическое правило.

Правило 2.4. При дополнении очередного отношения к формируемому *контексту*, прежде всего, выбираем отношения, которые являются подчиненными к уже выбранным отношениям.

В качестве промежуточной модели данных используется *таблица связанных соединений*.

Рассмотрим преобразование представления реляционной БД со схемой: $R_1, R_2, ..., R_k$ в *таблицу связанных соединений* (C,l), где C – схема отношения, определенная на множестве атрибутов $A_1, A_2, ..., A_n$, l – вектор вхождения длины k.

Определим принцип формирования кортежей $t \in c$, где c – реализация (множество кортежей) схемы отношения C. Рассмотрим все возможные сочетания без повторений отношений $R_1, R_2, ..., R_k$, удовлетворяющие свойству СБПИ. Пусть $P'=\{R_{m(1)}, R_{m(2)}, ..., R_{m(s)}\}$ – текущее сочетание отношений и p' его реализация, ограниченная логической формулой F: $p'=\sigma_F(R_{m(1)} \bowtie R_{m(2)} \bowtie \ldots \bowtie R_{m(s)})$.

Для каждого кортежа $u \in p'$ формируем кортеж t по следующим правилам: $t[A_j]=u[A_j]$, если атрибут A_j принадлежит соединению, и $t[A_j]=emp$ в противном случае, где emp – пустое значение. Каждому кортежу поставим в соответствие битовый вектор $l(t) = (l_1(t), l_2(t), ..., l_k(t))$, где $l_j(t)=1$, если реализация r_j схемы R_j участвует в текущем соединении, и $l_j(t)=0$ в противном случае.

Рассмотрим отношение частичного порядка над кортежами $t \in c$.

Определение 2.6. Кортеж $t \in c$ является менее определенным или равным кортежу $t' \in c$, когда для любого атрибута A_i выполнено: если $t[A_i] \neq t'[A_i]$, то $t[A_i]=emp$ и $l_j(t') \geq l_j(t)$, $j=1,\ldots,k$. В этом случае будем писать: $t \prec t'$ и назовем кортеж t подчиненным кортежу t'.

В представлении c достаточно хранить только кортеж t', который содержит в себе все менее определенные либо равные кортежи. Следовательно, завершающим этапом построения представления c является удаление в нем всех подчиненных кортежей.

Определение 2.7. Соединение отношений, удовлетворяющих свойству СБПИ будем называть связанным соединением.

Пусть $X(J)=([R_{j(1)}] \cup [R_{j(2)}] \cup \ldots \cup [R_{j(m)}])$, где $J=(j(1),j(2),\ldots,j(m))$, и $[R_{j(i)}]$ – множество атрибутов отношения $R_{j(i)}$. Определим операцию проекции на множестве c.

Определение 2.8. $\pi_{X(J)}(c)$ есть совокупность кортежей $u[X(J)]$, определенных на множестве атрибутов $X(J)$, где для каждого $u[X(J)]$ существует кортеж $t \in c$ такой, что $u[X(J)]=t[X(J)]$ и $l_{j(i)}(t)=1$, $i=1,2,\ldots,m$.

Основываясь на способе формирования таблицы c, сформулируем ее важные свойства.

Теорема 2.1. Для любого множества отношений $R'=\{R'_1, R'_2, ..., R'_s\}$, удовлетворяющего свойству СБПИ, выполнено:

$$\pi_{R'}(c)=\sigma_F(R'_1 \bowtie R'_2 \bowtie \ldots \bowtie R'_s).$$

Теорема 2.2. Представление c всегда существует и единственно для любой схемы реляционной БД.

В третьей главе «Модель данных «композиционная таблица»» рассматривается построение представления *композиционной таблицы*.

Обозначим $R_1, R_2, ..., R_k$, – исходные реляционные отношения, C – соответствующая этим отношениям *таблица связанных соединений*, R^* – результирующая таблица.

Пусть X, Y_i , Z_i – множества атрибутов из R $(i=1,2,\ldots, N)$. Атрибуты X остаются неизменными в R^* и являются наименованиями строк, значения атрибутов Y_i становятся именами столбцов в R^*, домены атрибутов Z_i, дополненные пустым значением, распределяются между доменами новых атрибутов, введен-

ных для значений Y_i. Естественными являются ограничения: $X \cap Y_i = \varnothing$, $X \cap Z_i = \varnothing$, $Y_i \cap Z_i = \varnothing$ $(i = 1, 2, \ldots, N)$. $|Dom(Y_i)| = L_i$, $|Z_i| = M_i$, где $Dom(Y_i)$ область значения атрибута Y_i в исходной БД.

Схема результирующего представления строится из исходных отношений по следующему правилу:

$$Sch(C) = \{X, Y_1, \ldots, Y_N, Z_1, \ldots, Z_N\} \Rightarrow$$

$$\Rightarrow Sch(CT) = \{X, \cup Dom(Y_i) \times \{Z_i\}\ (i = 1, 2, \ldots, N)\},$$

где Sch – схема описания отношения, Dom – область значений атрибута, $Dom(Y_i) = Dom(Y_{i1}) \times Dom(Y_{i2}) \times \ldots$, $Y_{ij} \in Y_i$.

В данной работе предлагается отказаться от необходимости выполнения функциональных зависимостей вида: $X, Y_i \rightarrow Z_i$, $i = 1, 2, \ldots, N$, что позволит иметь в одной ячейке гиперкуба несколько значений (список) атрибутов Z_i.

Определение 3.1. Множество атрибутов KZ_{jp} будем называть ключом атрибута $Z_{jp} \in Z_j$ в *контексте* P, если $KZ_{jp} \subseteq [P]$, зависимость $KZ_{jp} \rightarrow Z_{jp}$ выводима в FD^0, и не существует выводимой в FD^0 зависимости $Y \rightarrow Z_{jp}$, где $Y \subset KZ_{jp}$ и FD^0 – множество функциональных зависимостей на атрибутах отношений из P.

Определение 3.2. Значение атрибута $t[Z_{jp}]$, где $Z_{jp} \in Z_j$, для текущего кортежа $t \in C$ дублирует значение $t'[Z_{jp}]$, $t' \in C$, если:

1) $t[Z_{jp}] = t'[Z_{jp}]$,
2) $t[X] = t'[X]$, $t[Y_j] = t'[Y_j]$,
3) $t[KZ_{jp}] = t'[KZ_{jp}]$.

Смысл определения 3.2 следующий: если в выбранном *контексте* есть отношение, в котором идентифицируется (функционально определяется) отдельное значение атрибута, то это значение является важным для приложения, и если оно совпадает с другим значением этого же атрибута, то это не будет дублированием. В противном случае в контексте приложения значения параметра интерпретируются как список возможных значений, тогда в списке не должно быть совпадающих значений.

Предполагается, что все одноименные атрибуты в БД являются однородными, то есть являются однотипными и описывают одну и ту же характеристику в прикладной области.

Определение 3.3. Значения $z_{jp}^{\,i}$ в ячейке $r_k^*[y_j.Z_{jp}]$ строки r_k^*, будем называть однородными, если $z_{jp}^{\,i} \in Dom(Z_{jp})\ \forall\ i$.

Определение 3.4. Представление r^* сформировано корректно, если:

1. В каждой ячейке r^* содержатся однородные значения.
2. В каждой ячейке r^* отсутствуют дублированные значения.
3. В каждой строке $r_1^* \in r^*$ с определенными значениями атрибутов из X, в ячейке $r_1^*[y_j.Z_{jp}]$, где y_j – определенные значения, $j = 1, \ldots, N$, $p = 1, \ldots, L_j$, выполнено:

   а) содержатся все значения, соответствующие наборам $r_1^*[X]$, y_j,

б) отсутствуют значения z_{jp}^{1}, для которых строка $(r_1^*[X], y_j, z_{jp}^{1})$, $z_{jp}^{1} \in r_1^*[y_j.Z_{jp}]$ не может быть получена при проекции связанного соединения некоторых отношений из набора $R_1, R_2, \ldots, R_M$ по атрибутам X, Y_j, Z_{jp}.

Теорема 3.1. Представление r^* всегда корректно и единственно для совокупности отношений $R_1, R_2, \ldots, R_M$, образующих связанные соединения.

Рассмотрены образы зависимостей ***DEP***, которые используются при установлении иерархий в измерениях и анализе корректности заполнения значений мер *композиционной таблицы*.

Показано, что образы функциональных зависимостей, введенные для представления *композиционной таблицы*, являются достаточными для выполнения функциональных зависимостей для исходного реляционного отношения.

Теорема 3.2. Пусть r произвольная реализация схемы R, удовлетворяющая функциональной зависимости, в правой части которой атрибут X_0, тогда:

1. (FXX). Если $X' \to X_0$ удовлетворяет r^*, то $X' \to X_0$ удовлетворяет r.

2. (FYX). Если для произвольных строк r_1^*, $r_2^* \in r^*$ из условия $r_1^*[y_{j1}.Z_{jp(1)}] \neq emp$, $r_2^*[y_{j2}.Z_{jp(2)}] \neq emp$ и $y_{j1}' = y_{j2}'$, $y_{j1}' \subseteq y_{j1}$, $y_{j2}' \subseteq y_{j2}$, y_{j1}', $y_{j2}' \in Dom(Y_j')$ $\forall$ j: $Y_j \supseteq Y_j'$, $Y' = \cup Y_j'$, следует $r_1^*[X_0] = r_2^*[X_0]$, тогда $Y' \to X_0$ удовлетворяет r.

3. (FZX). Если из условия $S_{jp(i)}^{1} \cap S_{jp(i)}^{2} \neq \varnothing$ $\forall$ $p(i)$, $i = 1, \ldots, l_j$, где $S_{jp(i)}^{1} = \cup r_1^*[y_{j1}^{k}.Z_{jp(i)}]$ $(k=1,\ldots,q_1)$, $S_{jp(i)}^{2} = \cup r_2^*[y_{j2}^{k}.Z_{jp(i)}]$ $(k=1,\ldots,q_2)$, $Z_{jp(i)} \in Z'$ для произвольных строк r_1^*, $r_2^* \in r^*$, следует $r_1^*[X_0] = r_2^*[X_0]$, тогда $Z' \to X_0$ удовлетворяет r.

Теорема 3.3. Пусть r произвольная реализация схемы R, удовлетворяющая функциональной зависимости, в правой части которой атрибут Y_0, тогда:

1. (FXY). Если для произвольных строк r_1^*, $r_2^* \in r^*$ из условия $r_1^*[X'] = r_2^*[X']$, $r_1^*[y_{j1}.Z_{jp(1)}] \neq emp$ и $r_2^*[y_{j2}.Z_{jp(2)}] \neq emp$ для некоторого j, следует $y_1^0 = y_2^0$, где $y_1^0 \in y_{j1}$, $y_2^0 \in y_{j2}$, и $y_1^0, y_2^0 \in Dom(Y_0)$, тогда $X' \to Y_0$ удовлетворяет r.

2. (FYY). Если для произвольных строк r_1^*, $r_2^* \in r^*$ из условия $r_1^*[y_{j1}.Z_{jp(1)}] \neq emp$, $r_2^*[y_{j2}.Z_{jp(2)}] \neq emp$ и $y_{j1}' = y_{j2}'$, $y_{j1}' \subseteq y_{j1}$, $y_{j2}' \subseteq y_{j2}$, y_{j1}', $y_{j2}' \in Dom(Y_j')$ $\forall$ j: $Y_j \supseteq Y_j'$, $Y' = \cup Y_j'$, следует $y_1^0 = y_2^0$, где $y_1^0 \in y_{k1}$, $y_2^0 \in y_{k2}$ для некоторого k, и y_1^0, $y_2^0 \in Dom(Y_0)$, тогда $Y' \to Y_0$ удовлетворяет r.

3. (FZY). Если из условия $S_{jp(i)}^{1} \cap S_{jp(i)}^{2} \neq \varnothing$ $\forall$ $p(i)$, $i = 1, \ldots, l_j$, где $S_{jp(i)}^{1} = \cup r_1^*[y_{j1}^{k}.Z_{jp(i)}]$ $(k=1,\ldots,q_1)$, $S_{jp(i)}^{2} = \cup r_2^*[y_{j2}^{k}.Z_{jp(i)}]$ $(k=1,\ldots,q_2)$, $Z_{jp(i)} \in Z'$ для произвольных строк r_1^*, $r_2^* \in r^*$, следует $y_1^0 = y_2^0$, где $y_1^0 \in y_{q1}^{b}$, $y_2^0 \in y_{q1}^{g}$ для некоторых b, g, q, и $y_1^0, y_2^0 \in Dom(Y_0)$, тогда $Z' \to Y_0$ удовлетворяет r.

Теорема 3.4. Пусть r произвольная реализация схемы R, удовлетворяющая функциональной зависимости, в правой части которой атрибут Z_0, тогда:

1. (FXZ). Если для произвольных строк r_1^*, $r_2^* \in r^*$ из условия: $r_1^*[X'] = r_2^*[X']$ и существуют y_{k1}, и y_{k2} такие, что $r_1^*[y_{k1}.Z_0] \neq emp$, $r_2^*[y_{k2}.Z_0] \neq emp$ для некоторого k, следует $r_1^*[y_{k1}.Z_0] = r_2^*[y_{k2}.Z_0]$, тогда зависимость $X' \to Z_0$ удовлетворяет r.

2. (FYZ). Если для произвольных строк r_1^*, $r_2^* \in r^*$ из условий $r_1^*[y_{j1}.Z_{jp(1)}] \neq emp$, $r_2^*[y_{j2}.Z_{jp(2)}] \neq emp$, $r_1^*[y_{k1}.Z_0] \neq emp$ и $r_2^*[y_{k2}.Z_0] \neq emp$ для

некоторого k и $y_{j1}'=y_{j2}'$, $y_{j1}'\subseteq y_{j1}$, $y_{j2}'\subseteq y_{j2}$, y_{j1}', $y_{j2}'\in Dom(Y_j')$ $\forall$ j: $Y_j\supseteq Y_j'$, $Y'=\cup Y_j'$, следует $r_1^*[y_{k1}.Z_0]=r_2^*[y_{k2}.Z_0]$, тогда зависимость $Y'\rightarrow Z_0$ удовлетворяет r.

3. (FZZ). Если из условия $S_{jp(i)}^{1}\cap S_{jp(i)}^{2}\neq\varnothing$ $\forall$ $p(i)$, $i = 1, ..., l_j$, где $S_{jp(i)}^{1}=\cup r_1^*[y_{j1}^{k}.Z_{jp(i)}]$ ($k=1,\ldots,q_1$), $S_{jp(i)}^{2}=\cup r_2^*[y_{j2}^{k}.Z_{jp(i)}]$ ($k=1,\ldots,q_2$), $Z_{jp(i)}\in Z'$ для произвольных строк r_1^*, $r_2^*\in r^*$, следует $r_1^*[y_{q1}^{b}.Z_0]=r_2^*[y_{q2}^{g}.Z_0]$ для некоторых b, g, q, тогда зависимость $Z'\rightarrow Z_0$ удовлетворяет r.

В трех последних теоремах раздела доказано, что образы зависимостей исходной базы данных являются структурными ограничениями на *композиционную таблицу*. В частности, если одна ячейка таблицы не пуста, то не пуста, связанная с ней зависимостью, другая ячейка таблицы, и наоборот.

В четвертой главе «Реализация программного обеспечения системы» описано программное обеспечение, используемое для формирования представления *композиционной таблицы*.

С помощью разработанного программного обеспечения пользователь осуществляет формирование схемы *композиционной таблицы*, иерархий в измерениях, логических ограничений на измерения, *контекстов измерений* и вывод *композиционной таблицы* на экран.

Система генерации *композиционной таблицы* реализована в среде разработки Delphi. Доступ к базе данных осуществляется с помощью библиотеки ADODB. Запросы к БД осуществляются с помощью команд языка SQL.

Выбор ADODB для работы с СУБД обусловлен универсальностью данной библиотеки. При использовании ADODB, для перехода с одной СУБД на другую нужно будет поменять только параметры соединения с базой данных. ADODB поддерживает практически все системы управления базами данных, используемые разработчиками для хранения информации.

Работа пользователя с разработанным программным обеспечением осуществляется в следующем порядке.

1. Формирование схемы *композиционной таблицы*.

Пользователь осуществляет формирование множества атрибутов X, множеств $Y_1, Y_2,\ldots, Y_N$ и соответствующих множеств мер $Z_1, Z_2,\ldots, Z_N$ из множества всех атрибутов из таблиц исходной базы данных. Производится проверка условий $X\cap Y_i=\varnothing$, $(X\cup Y_i)\cap Z_i=\varnothing$, $i=1,2,\ldots N$.

2. Формирование иерархий измерений.

Пользователь имеет возможность определить желаемый порядок уровней иерархии для измерений $X, Y_1, Y_2,\ldots, Y_N$. Далее осуществляется автоматическое формирование схемы иерархии, при котором используются функциональные и многозначные зависимости исходной базы данных и порядок атрибутов, заданный пользователем. Затем пользователю предоставляется возможность модифицировать иерархии.

3. Задание логических ограничений на измерения.

Пользователь осуществляет задание логических ограничений на измерения $F_0(X)$, $F_1(Y_1)$, $F_2(Y_2),\ldots, F_N(Y_N)$. $F_i=F_{i1}\vee F_{i2}\vee\ldots\vee F_{is(i)}$, где F_{ij} – атомарные условия: $A_l\Theta const$ либо $A_l\Theta A_m$, где Θ – операция ($=$, $\neq$, $\leq$, $\geq$, $<$, $>$). Атомарные условия могут быть заданы как на атрибутах, входящих в схему *композиционной*

таблицы, так и на любых атрибутах из таблиц исходной базы данных. Затем осуществляется автоматическое формирование *контекста приложения* $\boldsymbol{C_{full}}$ по атрибутам $\boldsymbol{X \cup \{Y_j\} \cup \{Z_j\}}$. Также формируются *контексты* для тех измерений, в логические формулы которых входят атрибуты, не принадлежащие отношениям $\boldsymbol{C_{full}}$, по атрибутам измерения и логической формулы для измерения.

4. Формирование *контекстов измерений*.

Пользователь имеет возможность выбрать, для каких измерений должен быть сформирован *контекст* либо *псевдоконтекст*. Формирование *контекстов* и *псевдоконтекстов* осуществляется автоматически по атрибутам измерения.

5. Формирование *композиционной таблицы*.

Осуществляется формирование *таблицы связанных соединений* с логическим ограничением $\boldsymbol{F(C)=F_0(X) \wedge (F_1(Y_1) \vee F_2(Y_2) \vee \ldots \vee F_N(Y_N))}$ и реализаций измерений, для которых сформирован *контекст* либо *псевдоконтекст*. Далее формируется представление *композиционной таблицы* и осуществляется ее вывод на экран.

Для формирования таблицы соединений используются вспомогательные алгоритмы: ***COMB*** – генерация сочетаний без повторений из $\boldsymbol{k}$ элементов по $\boldsymbol{m}$; ***IsSSBP*** – проверка выполнения свойства СБПИ; ***JoinTable*** – формирование текущего соединения с преобразованием его в $\boldsymbol{C}$–таблицу. Описание указанных алгоритмов приведено во второй главе диссертации.

Вывод на экран *композиционной таблицы* осуществляется при выполнении алгоритма ***LOADR***.

В пятой главе «Описание и реализация приложения» представлена экспериментальная проверка разработанной технологии аналитической обработки данных в условиях кардиологического диспансера для осуществления дифференциальной диагностики пациентов.

Произведено построение шкалы оценки тяжести артериальной гипертензии на основе анализа данных, представленных в виде *композиционной таблицы*. Для этого потребовалась аналитическая обработка данных из выписок пациентов.

Исходные данные представляли собой выписки из истории болезни пациентов кардиологического диспансера в формате электронного документа. Было реализовано дополнительное программное обеспечение для получения необходимой информации из необработанных данных, описанное в приложении. На основе данных из выписок пациентов была построена схема базы данных «Кардиологический диспансер» и реализована соответствующая реляционная база данных.

Для решения задачи диагностирования пациентов кардиологического диспансера было построено следующее представление *композиционной таблицы*, используя программное обеспечение, описанное в главе 4:

Атрибуты множества $\boldsymbol{X}$: № истории болезни;

Атрибуты множества $\boldsymbol{Y_1}$: вид обследования, численный показатель;

Атрибуты множества $\boldsymbol{Z_1}$: значение численного показателя;

Традиционно для расчета диагностической шкалы используется решающая функция: $F(x)=a_1x_1+a_2x_2+\ldots+a_Nx_N$, где $x=(x_1,x_2,\ldots,x_N)$ – вектор значений выделенных параметров (координат в пространстве параметров), $a=(a_1,a_2,\ldots,a_N)$ – веса выделенных параметров (коэффициенты).

Для значений функции $F(x)$ определяются границы (оценочная шкала): $g_0, g_1, \ldots, g_K$, где K – количество групп объектов $O_1, O_2, \ldots, O_K$. При условии, что $g_0<g_1<\ldots<g_K$, определение принадлежности произвольного объекта o с вектором значений параметров x' к группе O_j сводится к проверке выполнения неравенства: $g_{j-1}<F(x')<g_j$. При выполнении равенства значения функции F какой-либо границе $F(x')=g_j$ возникает ситуация неопределенности.

Для определения значений коэффициентов $(a_1,a_2,\ldots,a_N)$ и значений границ $g_0, g_1, \ldots, g_K$, в распознавании образов традиционно используются обучающие выборки, заданные множеством групп объектов $O_1, O_2, \ldots, O_K$. Пусть объект $o_{ij}\in O_i$ характеризуется вектором значений параметров: $x_{ij}=(x_{ij1},x_{ij2},\ldots,x_{ijN})$. Функционалом риска выбрано суммарное количество ошибок E при отнесении объекта к группе.

Таким образом, задача построения оценочной шкалы может быть записана в следующем виде: $E \rightarrow min$, $g_0<g_1<\ldots<g_K$, $-1 \le a_i \le 1$, $i=1,2,\ldots,N$. Ограничения на коэффициенты a_i реализуются за счет масштабирования. Заметим, что целевая функция (функционал риска) не является дифференцируемой. Это исключает использование градиентных методов для решения поставленной задачи.

Все пациенты кардиологического диспансера в соответствии с врачебным заключением были разделены на 3 группы: I, II и III степени заболевания.

Предварительный анализ данных заключался рассмотрении гистограмм распределения числовых параметров из выписок. Этот анализ позволил выделить наиболее значимые параметры. В результате предварительного анализа было выбрано 14 параметров: Частота сердечных сокращений (ЧСС), Систолическое артериальное давление (САД), Диастолическое артериальное давление (ДАД), Триглицериды, Холестерин, Hb, СОЭ, Удельный вес мочи (УВМ), Размер аорты (АО), ЗСЛЖ, КСР, ЛП_ЭХО-КГ, МЖП, Возраст.

При поиске минимума функционала риска было выяснено, что неплохое приближение к оптимальным значениям весов параметров дает значение информативной меры Кульбака соответствующего параметра. Значение меры, деленное на максимальное значение параметра (нормировка) и использованное в качестве начального приближения, дает решения, близкие к оптимальному.

На некотором интервале в области оптимальности функционал риска имеет постоянное значение. Кроме того, вблизи оптимума имеются локальные оптимальные значения. Следовательно, окончательный выбор оптимального значения веса параметра целесообразно сделать вручную. Для повышения устойчивости решения этот оптимум целесообразно выбрать из середины интервала, где функционал риска имеет наименьшее постоянное значение.

В результате выполненных расчетов получены веса параметров и границы.

На разработанное программное обеспечение «Электронная шкала оценки тяжести и мониторинга артериальной гипертензии» получено свидетельство о регистрации разработки.

В заключении предложено изложение основных результатов, полученных в диссертационной работе.

В приложениях описана обработка неформализованных данных в выписках пациентов кардиологического диспансера и построение схемы базы данных для хранения данных по пациентам, приведены гистограммы распределения числовых параметров из выписок пациентов кардиологического центра. Для двух алгоритмов формирования контекстов приведено сравнение по количеству итераций.

Основные научные результаты

1. Разработаны модель и алгоритмы формирования *композиционной таблицы*.
2. Исследованы свойства и условия существования промежуточного представления данных – *таблицы связанных соединений*.
3. Разработан алгоритм автоматизированного формирования иерархий в измерениях.
4. Разработан алгоритм автоматического формирования *контекстов измерений* и *контекста приложения* и доказана корректность построения представления *композиционной таблицы*.
5. Реализовано программное обеспечение, формирующее представление *композиционной таблицы*, и на его основе разработана диагностическая шкала оценки тяжести артериальной гипертензии.

Публикации по теме диссертации

Статьи, опубликованные в журналах из списка ВАК

1. Редреев П.Г. Построение табличных приложений со списочными компонентами // Информационные технологии. 2009. №5. С. 7-12.
2. Редреев П.Г. Построение иерархий в многомерных моделях данных // Известия Саратовского университета. Серия Математика. Механика. Информатика. 2009. Т. 9. вып. 4. ч. 1. С. 84-87.
3. Зыкин С. В., Редреев П. Г., Чернышев А. К. Формирование представлений данных для построения медицинских диагностических шкал // Омский научный вестник. Серия Приборы, машины и технологии. 2011. № 2 (100). С. 160-165.

Другие публикации

4. Редреев П.Г. Формирование модели данных со списочными компонентами для работы с реляционными базами данных по технологии OLAP // Материалы XLVI международной научной студенческой конференции «Студент и научно-технический прогресс»: Информационные технологии. Новосибирск. 2008. С. 170.
5. Редреев П.Г. Формирование иерархий измерений многомерных моделей данных // Седьмая международная конференция «Перспективы систем

информатики»: материалы рабочего семинара «Наукоемкое программное обеспечение». Новосибирск. 2009. С. 231-234.
6. Редреев П.Г. Формирование представления данных со списочными компонентами для работы с реляционными базами данных по технологии OLAP // Материалы Всероссийской конференции с международным участием «Знания – Онтологии – Теории» (ЗОНТ-09). Новосибирск. 2009. Т.1. С. 232-235.
7. Редреев П.Г. Автоматизация построения иерархий в измерениях многомерных моделей данных // Новые алгебро-логические методы решения систем уравнений в алгебраических системах. Тезисы докладов. Омск. 2009. С. 58-59.
8. Зыкин С.В., Редреев П.Г., Полуянов А.Н., Чернышев А.К., Колмогорова О.Н. Электронная шкала оценки тяжести и мониторинга артериальной гипертензии // Хроники объединенного фонда электронных ресурсов «Наука и образование». №2 (21). 2011. URL: http://ofernio.ru/portal/newspaper/ofernio/2011/2.doc. (дата обращения: 01.03.2011)

Работа выполнялась при поддержке Российского фонда фундаментальных исследований (проект № 09-07-00059-а).

Редреев Павел Григорьевич

Разработка и исследование обобщенной табличной модели данных со списочными компонентами

Автореферат

диссертации на соискание ученой степени

кандидата физико-математических наук

Подписано в печать ____ 2011
Формат 60x84 1/16. Бумага офсетная.
Печать офсетная. Усл. печ. л. 1,0. Уч.-изд. л. 1,2.
Тираж 120 экз.

Издательство