На правах рукописи

Масленников Виталий Александрович

ИНТЕГРИРОВАННЫЕ ОБЪЕКТНО-РЕЛЯЦИОННЫЕ ЛОГИЧЕСКИЕ МОДЕЛИ ПРЕДСТАВЛЕНИЯ ДАННЫХ ДЛЯ СИСТЕМ ПОТОКОВОЙ ОБРАБОТКИ ИНФОРМАЦИИ

Специальность 05.13.11

Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени кандидата технических наук

Уфа 2008

Работа выполнена на кафедре информатики
Уфимского государственного авиационного технического университета

Научный руководитель д-р техн. наук, проф.
Кабальнов Юрий Степанович

Официальные оппоненты д-р техн. наук, проф.
Павлов Сергей Владимерович

канд. техн. наук, доц.
Набатов Александр Нурович

Ведущее предприятие: ООО «Новойл-Автоматика», г. Уфа

Защита состоится «____» _____ 2008 г. в __ часов
на заседании диссертационного совета Д-212.288.07
при Уфимском государственном авиационном техническом университете
по адресу: 450000, г. Уфа, ул. К. Маркса, 12

С диссертацией можно ознакомиться в библиотеке университета

Автореферат разослан «_____» _____ 2008 г.

Ученый секретарь
диссертационного совета
д-р техн. наук, проф. **С.С. Валеев**

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы

В настоящее время широкое распространение получили большие базы данных (БД) в системах потоковой обработки информации, среди которых можно отметить БД билинговых систем, БД геоинформационных систем и БД крупных торговых сетей. Характерными особенностями больших БД в системах потоковой обработки информации, как правило, являются большой объем хранимой и поступающей информации, интерактивный многопользовательский режим работы, нестационарность схемы данных БД, динамические методы обработки данных, сложная пред- и постобработка потока поступающих данных.

Для построения подобных БД, как правило, применяют классический реляционный подход или современный объектно-ориентированный. При построении подобных БД для систем потоковой обработки информации с использованием реляционного подхода возникает ряд трудностей, вызванных структурной избыточностью логической модели данных; неструктурированным хранением алгоритмов обработки данных; сложностью организации многопользовательских интерактивных режимов для аналитической обработки данных (первичный Data-mining); нарушением целостности БД в процессе обновления схемы данных и обработки данных вследствие сложности установления связей между группами сущностей. Кроме того, при использовании объектно-ориентированного подхода возникают трудности, связанные отсутствием строгого математического аппарата, позволяющего строить логические модели данных и реализовывать операции их обработки, а также с низкой скоростью поиска в больших массивах данных, что затрудняет использование подобных БД в режиме реального времени.

Подходы к решению подобных задач рассматриваются в работах А.С.Усова, Г. Буча.

В работе предлагается новый тип логических моделей представления данных, а именно интегрированных объектно-реляционных моделей данных и основанные на них методы структуризации алгоритмов обработки и управления данными.

Цель диссертационной работы

Целью работы является разработка интегрированных объектно-реляционных логических моделей представления данных для больших реляционных баз данных в системах потоковой обработки информации, способных сохранять целостность данных в условиях значительных динамических изменений схемы данных и методов их обработки.

Задачи исследования

Для достижения цели работы поставлены и решены следующие задачи:

1. Разработать интегрированную объектно-реляционную логическую модель представления данных большой реляционной БД, позволяющей: снизить структурную избыточность логической модели представления данных; предотвратить появление структурных аномалий данных в условиях динамичного изменения логической схемы данных и самих данных.

2. Разработать объектно-реляционную модель хранения алгоритмов обработки данных, позволяющих упорядочить их хранение, и обеспечить управление процессом обработки данных.

3. Разработать критерии качественной оценки логических моделей представления данных БД для систем потоковой обработки данных.

4. Оценить эффективность предложенных моделей данных и алгоритмов на примере БД системы потоковой обработки складских документов крупной торгово-сервисной сети.

Методы исследования

При решении поставленных задач в работе использовались теория реляционных баз данных, теория объектно-ориентированного подхода, а также применение реляционной алгебры и теории множеств. Экспериментальная проверка теоретических результатов проводилась на основе системы потоковой обработки складских документов крупной торгово-розничной сети ТРК «Июнь» г. Санкт- Петербург.

Основные научные результаты, выносимые на защиту:

- интегрированная объектно-реляционная логическая модель представления данных большой реляционной БД;
- объектно-реляционная модель хранения алгоритмов обработки данных;
- критерии качественной оценки логических моделей представления данных БД для систем потоковой обработки данных;
- результаты экспериментальной проверки эффективности предложенных моделей представления данных и хранения алгоритмов.

Научная новизна работы:

1. Применение объектно-реляционного подхода к построению логических моделей представления данных и алгоритмов их обработки для больших баз данных, в отличие от известных, позволяет объединить в рамках единой иерархической структуры как логическую структуру данных, так и структуру алгоритмов обработки данных. Это обеспечивает в условиях динамического изменения схемы данных и методов их обработки снижение структурной и алгоритмической избыточности и обеспечивает эффективное управление процессом обработки данных.

2. Предложен новый метод преобразования классической реляционной логической модели представления данных в объектно-реляционную логическую модель представления данных путем объединения семантически подобных элементов в исходной реляционной логической модели данных, что позволяет снизить количество потенциальных структурных аномалий.

3. Предложены критерии качественной оценки логических моделей представления данных БД, отражающие их структурную сложность и функциональную надежность БД, позволяющие производить количественное сравнение структурной сложности логических моделей представления данных и осуществлять целенаправленное изменение модели с целью получения ее наилучших характеристик.

Обоснованность и достоверность результатов диссертации

Обоснованность результатов, полученных в работе, базируется на использовании апробированных научных положений и методов исследования, согласованности экспериментальных результатов с теоретическими исследованиями.

Достоверность полученных результатов и выводов подтверждается результатами проведенных численных экспериментов.

Практическая значимость результатов

Предложенная интегрированная объектно-реляционная логическая модель представления данных и алгоритмы на основе разработанного метода преобразования классической реляционной логической модели представления данных в объектно-реляционную использовались при разработке системы потоковой обработки складских документов крупной торгово-розничной сети ТРК «Июнь» г. Санкт- Петербург. Это позволило повысить бесперебойность выдачи информации в режиме реального времени менеджменту компании о текущем объеме продаж, остатке товара, обеспечить актуальность и достоверность получаемой информации, а также снизить сложность системы в 1,4 раза.

Публикации

По материалам диссертации опубликовано 11 печатных работ (в том числе 1 статья в рецензируемом журнале из списка ВАК), 3 доклада в сборниках трудов конференций, 1 монография (в соавторстве), список которых приведен в конце автореферата.

Структура и объем работы

Диссертационная работа состоит из введения, четырех глав, заключения, библиографии. Основная часть содержит 116 страниц и включает в себя 47 рисунков и 6 таблиц. Список литературы содержит 124 наименования.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обосновывается актуальность задачи, решаемой в диссертационной работе, формулируются задачи и цели исследования, отмечается научная новизна и практическая ценность полученных результатов.

В первой главе рассматриваются особенности работы БД потоковой обработки информации и сложности построения логической модели представления данных.

Основными особенностями БД потоковой обработки являются отсутствие формализованной модели системы, зашумленность, нестационарность, невоспроизводимость эксперимента.

Как известно, БД обладает рядом следующих основных параметров, изменяющихся в процессе ее функционирования характерными чертами организационного объекта, наиболее важными из которых являются эволюция структуры, цели и задачи системы, непрерывное изменение количества ее элементов (таблиц, связей, алгоритмов обработки данных), что, в свою очередь, оказывает влияние на функциональность всей БД.

Особенностью крупного предприятия, является постоянное развитие его структуры. Это связано со многими факторами, самыми главными из которых являются работа предприятия во внешнем экономическом пространстве, существование предприятия в правовом пространстве, существование предприятия в пространстве постоянно изменяющейся технической базы (оснастки), влияние эволюции предприятия на административную подсистему.

В БД потоковой обработки данных объект реального мира не может быть описан в рамках одного кортежа или одного отношения, так как объект предметной области в логической модели данных представляется при помощи ряда записей нескольких реляционных отношений, связанных друг с другом. Для примера рассмотрим приходный документ. В его составе можно выделить такие отношения как заголовок, спецификация (многострочное приложение), стадии документа (этапы пути обработки документа), где в каждом из отношений может присутствовать от десяти до нескольких сотен записей. Описание одного объекта реального мира в БД потоковой обработки данных можно представить в виде понятия «бизнес объекта» (БО). БО – это элемент БД, который представляет собой множество отношений, связанных друг с другом. Исходя из выше сказанного, можно представить БО как множество таблиц и связей между ними: БО = $< T_{БО}, F_{БО}>$, где $T_{БО}=\{t_1, \ldots, t_{m_{БО}}\}$ – множество таблиц, $F_{БО}= \{f_1, \ldots, f_{n_{БО}}\}$ – множество связей, где $m_{БО}, n_{БО} \in [1; \infty)$.

Таким образом, можно ввести следующую меру сложности БО – сложность, как функция от количества таблиц, связей и атрибутов $S = f(T, F, A)$, где A – множество атрибутов.

Следует отметить, что функционирование БД потоковой обработки данных обусловлено большим количеством и сложностью структуры алгоритмов обработки данных. Использование оперативных транзакций требует выполнения аналитических запросов низкой и средней сложности.

Одним из решений данной задачи является использование *OLAP – OLTP* систем. Препятствием применения комбинации *OLAP – OLTP* систем в БД потоковой обработки информации является наличие сложных методов управления и большого количества сложных алгоритмов, состоящих из тесно взаимосвязанных частей.

Работа любой БД ведется в рамках транзакций, которые являются атомарными и неделимыми операциями работы с наборами данных. При обработке данных в БД неизбежно приходится сталкиваться с блоками информации сложной структуры. В реляционной системе можно выделить 4 простые операции, которые выполняются целостно (вставка, удаление, обновление, выборка (*insert, update, delete, select*)). При работе со сложно связанными друг с другом данными БД потоковой обработки информации невозможно обеспечить целостность данных, используя только элементарные транзакции. Каждая операция над объектом, хранимым в БД потоковой обработки информации, состоит из множества элементарных транзакций. Предлагается ввести понятие «бизнес транзакция» (БТ). БТ может использоваться в качестве количественной характеристики описательной сложности: $S_{БТ} = f(K_{БТ}, S_{NБТ})$, где $K_{БТ}$ - количество эле-

ментарных транзакций, выполняемых в рамках БТ, $S_{NБТ}$ - некая усредненная сложность, которая равна количеству кортежей в БТ.

Концептуально логическую модель представления данных в системе потоковой обработки информации можно представить в виде 4-х абстрактных классов, что позволяет использовать внутриклассовую общность и подобие сущностей. Следующей ступенью унификации является разработка логической модели представления данных БД потоковой обработки информации.

Концептуальная логическая модель представления данных изображена на рисунке 1.1.

Как видно из рисунка, логическая модель представления данных системы потоковой обработки складских документов состоит из следующих элементов:

1. *Справочники*. В качестве справочников могут выступать таблицы, содержащие любые перечислимые данные, являющиеся основой для построения документов. В качестве примера можно привести такие справочники, как: «Пользователи», «Склады», «Контрагенты», «Месяца» и т.д. Справочники могут ссылаться на другие справочники, но количество таких ссылок невелико.

2. *Документы*. Являются основой для построения системы учета. Именно на основе документы осуществляют те операции, которые требуют учета. На основе документов происходит «движение» товарно-денежных средств. Документы в основном состоят из ссылок на справочники, и именно ссылки документов составляют большую часть всех ссылок.

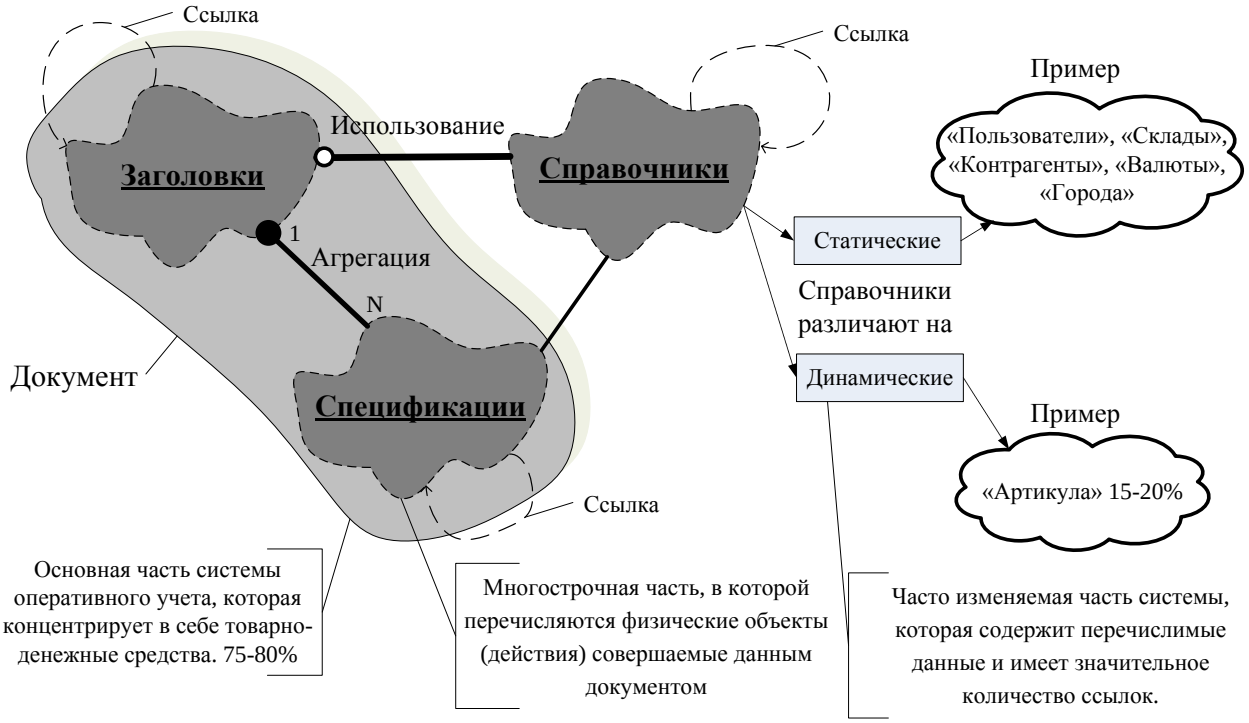

Рисунок 1.1 – Концептуальная логическая модель представления данных системы потоковой обработки складских документов

Документы, как правило, состоят из двух частей:

a) *заголовки* – общие заголовки документов;

b) *спецификации* – многострочная часть, в которой перечисляются физические объекты/действия, совершаемые на основе данного документа.

Могут существовать также спецификации, не имеющие заголовков. Как правило, это потоковые данные (данные с датчиков информации, электронные письма и т.п.).

Следует отметить, что основная масса данных сосредоточена в «Документах» и большая часть алгоритмов обрабатывает и анализирует именно данные, хранящиеся в «Документах».

Выполнен анализ недостатков применения классического реляционного подхода при создании логических моделей БД потоковой обработки информации. Отмечены следующие недостатки: сложность при реализации функциональной зависимости к группе сущностей; значительные затраты как человеческих, так и временных ресурсов на внесение изменений как в логическую модель представления данных, так и в алгоритмы обработки информации; невозможность использования реляционной модели данных для алгоритмического отображения процессов предметной области; трудоемкость при получении аналитических данных низкой и средней сложности для решения задач оперативного управления; низкая производительность из-за наличия структурной и алгоритмической избыточности, а также процедурной логики.

Рассмотрены основные недостатки применения объектно-ориентированного подхода при создании логических моделей представления данных. Среди них следует отметить инкапсуляцию свойств объекта, недостатки формализации модели данных, технику реализации хранения поведения объекта, а не данных об объекте.

Предлагается использовать объектно-реляционный подход при построении логических моделей данных для БД потоковой обработки информации, используя преимущества ООП и классического реляционного подхода. К таковым относятся четкий математический базис реляционного подхода (большинство современных СУБД имеют реляционной базис); возможность простой реализации функциональной зависимости группы сущностей (ООП); стройная иерархия структуры данных и алгоритмов их обработки (ООП); простота описания; высокая повторная используемость кода; возможность анализа системы на разных уровнях абстракции (ООП).

Во второй главе рассматриваются особенности применения объектно-ориентированного подхода при построении логических моделей представления данных в БД потоковой обработки информации при сохранении реляционного базиса. Анализ характерных особенностей БД, в которой происходит потоковая обработка набора объектов, показал, что для данного типа объектов свойственна иерархическая структура данных, функциональной особенностью которой является непрерывное эволюционное развитие. Следует уточнить, что рассматриваемая в данной работе логическая модель представления данных, хоть и базируется на классической объектно-ориентированной модели, но не тождественна ей. Она имеет множество отличий, обусловленных ее реализацией в контексте реляционных баз данных, а не языков программирования.

Так, одним из наиболее существенных отличий является характер работы с объектами в предлагаемой модели. Если в классической объектно-ориентированной модели, сам объект был активной единицей, то в предлагае-

мой логической модели представления данных такой активной единицей становится класс.

Кроме того, особенностью предлагаемого подхода является то, что объектно-реляционная база данных (ОРБД) хранит не просто различные объекты, принадлежащие различным классам, а множества объектов – их списки, и любая работа в ОРБД протекает не с отдельными объектами, а с их определенной выборкой, подсписком. Обращение к свойствам объектов, вызов их методов – все это имеет массовый характер, работа с одиночным объектом является исключением, частным случаем работы с множеством.

В предлагаемой логической модели представления данных класс является активной составляющей, способной в процессе работы оказывать индетерминированное влияние на свои экземпляры (объекты). К примеру, в БД не может существовать двух одинаковых объектов данного класса. Само поведение системы выражается через поведение классов, а не объектов, так как класс, при появлении нового своего экземпляра может изменять другие свои экземпляры по определенным правилам. Объекты лишены всякой самостоятельности и являются просто контейнерами для хранения свойств, делегируя свою активность классам и их методам. Следует отметить ключевую особенность предлагаемого подхода, заключающуюся в сохранении четко формализованного реляционного базиса. Поэтому данную логическую модель предполагается называть не объектно-ориентированной, а объектно-реляционной (ОР).

Исходными данными для метода построения логической ОР-модели представления данных является множество отношений $T=\{t_1, \ldots, t_n\}$, причем каждое из отношений t_i имеет множество атрибутов $A_i=\{a_1^i, \ldots, a_{k_i}^i\}$ где n – общее число отношений (таблиц в БД), k_i – число атрибутов в отношении t_i.

Суть метода заключается в последовательном выполнении следующей последовательности действий:

1) семантически подобные атрибуты (то есть атрибуты, которые несут одинаковую смысловую нагрузку в различных отношениях) объединяются в одно множество; из всех таких множеств формируется множество $A^c = \{a_1^c, \ldots, a_i^c, \ldots, a_l^c\}$, где a_i^c – множество семантически подобных атрибутов, l – количество семантически различных атрибутов в БД;

2) каждому множеству атрибутов a_i^c соответствует множество отношений $T_i^c \subseteq T$, содержащих атрибуты данного множества a_i^c, то есть $a_i^c \subseteq \bigcup_{j:\, t_j \in T_i^c} A_j$, при этом каждый атрибут из множества a_i^c принадлежит только одному отношению из множества T_i^c и такое отношение T_i^c обязательно существует, то есть $T^i \neq \varnothing$; из всех таких множеств $T_i^c \subseteq T$ формируется множество $T^c = \{T_1^c, \ldots, T_i^c, \ldots, T_l^c\}$.

3) Некоторые множества $T_i^c \in T^c$ могут совпадать, так как некоторым множествам семантически подобных атрибутов a_i^c могут соответствовать одинаковые по составу множества отношений; после замены таких одинаковых по составу множества отношений одним множеством получаем множество $C = \{C_i\}$, где C_i – класс-кандидат;

4) Между классами-кандидатами устанавливаются отношения иерархии, которые можно задать в виде матрицы E с элементами:

$$e_{ij} = \begin{cases} 1, \textit{если класс } C_j \textit{ является предком класса } C_i, i \neq j; \\ 0, \textit{в противном случае.} \end{cases}$$

с учетом этого можно представить структуру наследования в виде таблицы 2.1.

Таблица 2.1. Связи между отношениями в дереве наследования

		Родительские отношения			
	-	T'_1	T'_2	T'_3	…
Отношения наследники	T''_1	-	1	0	…
	T''_2	1	-	1	…
	T''_3	0	0	-	…
	…	…	…	…	…

Представлен пример, исходными данными для которого выбраны четыре сущности (рисунок 2.1):

Склад_ВхДок	
PK	**Ид_ВхДок**
	Ид_Контр_Ист
	Ид_Склад_Назн
	Док_Ном
	Док_Дата
	Сумма
	Выписка

Склад_ИсхДок	
PK	**Ид_ИсхДок**
	Ид_Контр_Назн
	Ид_Склад_Ист
	Док_Ном
	Док_Дата
	Сумма
	Ид_Довер

Склад_Возврат	
PK	**Ид_Возврат**
	Ид_Контр_Ист
	Док_Ном
	Док_Дата
	Ид_Склад_Назн
	Выписка
	Сумма

Склад_Списание	
PK	**Ид_Списание**
	Ид_Контр_Назн
	Док_Ном
	Док_Дата
	Ид_Склад_Ист
	Ид_Довер
	Сумма

Рисунок 2.1 – Исходное количество сущностей с атрибутами

В соответствии с методом построения логической ОР-модели представления данных получаем изображенную на рисунке 2.2, иерархическую логическую ОР-модель представления данных в виде UML схемы. В результате преобразований было получено 8 классов, хотя в исходной форме мы имеем всего 4 отношения. Для того чтобы можно было однозначно идентифицировать принадлежность объектов к определенному классу, этот класс должен существовать явно, пусть и фиктивным образом, как в данном случае, т.к. класс «Склад_ИсхДок» не содержит собственных атрибутов, а только интегрирует в себе атрибуты двух родительских классов.

Таким образом, можно вывести правило, что все сущности также в обязательном порядке заносятся в таблицу кандидатов классов, даже если у них отсутствуют собственные уникальные атрибуты.

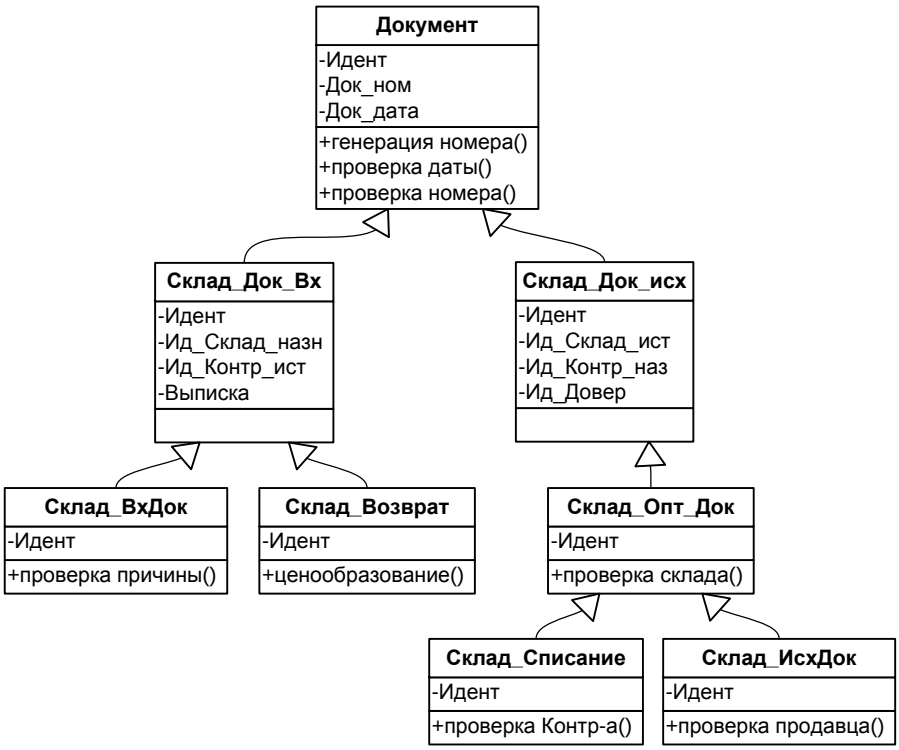

Рисунок 2.2 – Граф наследования реляционных отношений

Предлагаемая ОР-модель позволяет обеспечить свойство надежного и эффективного эволюционного развития при сохранении функциональности в интерактивном режиме работы, а также наделить логическую модель данных возможностью множественного наследования, что сложно реализовать в реляционной БД. Реализация классовой парадигмы ООП позволила сохранить реляционный базис БД, сохранить нормализованное состояние структуры данных, обеспечивая целостное состояние в условиях непрерывного развития.

Предложен метод грануляции семантически подобных реляционных алгоритмов (эти алгоритмы обеспечивают смысловую обработку данных в различных сущностях) в классовую структуру, для структуризации и наиболее точного соответствия бизнес логики предметной области.

Исходными данными для метода является множество алгоритмов $\alpha = \{\alpha_1, \ldots, \alpha_\gamma\}$ БД, где γ - число алгоритмов.

Метод аналогичен методу построения логической ОР-модели представления данных и включает в себя следующие основные процедуры:

1) семантически подобные алгоритмы (то есть алгоритмы, которые выполняют близкую по смыслу обработку данных в различных сущностях) объединяются в одно множество; из всех таких множеств формируется множество $\alpha^c = \{\alpha_1^c, \ldots, \alpha_i^c, \ldots, \alpha_m^c\}$, где α_i^c – множество семантически подобных с, m – количество семантически различных алгоритмов в БД;

2) каждому множеству алгоритмов α_i^c соответствует множество сущностей $T_i^c \subseteq T$, содержащих алгоритмы данного множества α_i^c, то есть

$\alpha_i^c \subseteq \bigcup_{j:\, t_j \in T_i^c} \alpha_j$, при этом каждый алгоритм из множества α_i^c принадлежит только одной сущности из множества T_i^c и такое отношение T_i^c обязательно существует, то есть $T^i \neq \varnothing$; из всех таких множеств $T_i^c \subseteq T$ формируется множество $T^c = \{T_1^c, ..., T_i^c, ..., T_l^c\}$.

3) Некоторые множества $T_i^c \in T^c$ могут совпадать, так как некоторым множествам семантически подобных алгоритмов α_i^c могут соответствовать одинаковые по составу множества отношений; после замены таких одинаковых по составу множества отношений одним множеством получаем множество $C = \{C_i\}$, где C_i – класс-кандидат;

4) составляется таблица конечных кандидатов, в соответствии с которой определяется, какая группа алгоритмов или отдельный алгоритм, принадлежит соответствующему классу (сущности).

Для соответствия предметной области предлагается произвести объединение структуры данных и структуры алгоритмов, так как они не могут существовать неразрывно и дополняют друг друга. В ее основе находится метакласс «Документ», который обслуживается тремя алгоритмами: генерация номера, проверка даты, проверка номера и т.д. (рисунок 2.2)

В третьей главе выполнен анализ критериев оценки качества БД как программного средства, которое выполняет роль централизованного хранилища информации, ее предобработки и анализа.

Предложено выполнять качественную оценку БД комплексным критерием сложности в соответствии с методами теории исследования сложных систем, предложенную Н.П. Бусленко. Сложностью системы, состоящей из элементов со сложностью S_i, где i = 1, …, n будем называть величину $S = \sum_{i=1}^{n} S_i \cdot r_i$, где r_i – число элементов i-го типа, входящих в систему.

Особенностями данной модели является то, что ссылочная сложность $S_{k1} >> S_{k2} >> S_{k3}$, где S_{k1} – ссылочная сложность между заголовками и справочниками, S_{k2} – ссылочная сложность между спецификациями и заголовками, S_{k3} – ссылочная сложность между справочниками и спецификациями. Для сравнительной оценки ссылочной сложности предлагаемой модели данных с существующей реляционной, необходимо выполнить процедуру типизации связей в БД.

Для комплексной оценки системы необходимо оценить сложность ее элементов. К основным элементам реляционной БД можно отнести: атрибуты, связи входящие и исходящие, и алгоритмы обработки данных.

Общая атрибутивная сложность системы можно определяется по следующей формуле:

$$A = \sum_{i=1}^{3} A_i \cdot N_i , \qquad (3.2)$$

где A_1 – количество атрибутов справочников, A_2 – количество атрибутов заголовков, A_3 – количество атрибутов спецификаций, N_1 – количество справочников в системе, N_2 – количество заголовков в системе, N_3 – количество спецификаций в системе.

Сложность связей входящих в систему определяется по следующим формулам:

$$S_{исх} = \sum_{i=1}^{3} S_{fk\,i} \cdot N_i , \qquad (3.3)$$

$$S_{вх} = N_3 + \sum_{i=1}^{3} S_{fk\,i} \cdot N_i , \qquad (3.4)$$

где S_{fk1} – количество вторичных ключей справочников, S_{fk2} – количество вторичных ключей заголовков, S_{fk3} – количество вторичных ключей спецификаций.

Общая сложность связей реляционной модели данных БД оценивается следующим образом: $S_{св} = S_{вх} + S_{исх}$. (3.5)

Алгоритмическую сложность системы необходимо оценивать по каждому метаклассу раздельно, это обусловлено тем, что сложность алгоритмов функционально зависит от количества вторичных ключей, атрибутов и бизнес - алгоритмов, которые обслуживают каждый из метаклассов системы. Для оценки сложности алгоритмов справочников используются следующие зависимости:

$$S_a^{спр} = S_{БА1} + (\sum_{i=1}^{2} k_i \cdot A_i) + k_3 \cdot \frac{S_{исх}}{N_1} , \qquad (3.6)$$

$$S_a^{шап} = S_{БА2} + k_1 \cdot A_2 + k_2 \cdot S_{fk2} + k_3 \cdot \frac{N_3}{N_2}, \qquad (3.7)$$

$$S_a^{спец} = S_{БА3} + k_1 \cdot A_3 + k_2 \cdot S_{fk3}, \qquad (3.8)$$

где $S_{БА1}$ – количество БА справочников, $S_{БА2}$ – количество БА заголовков, $S_{БА3}$ – количество БА спецификаций, k_1 – коэффициент вовлеченности атрибутов в алгоритмы справочников, k_2 – коэффициент вовлеченности атрибутов в алгоритмы заголовков, k_3 – коэффициент вовлеченности атрибутов в алгоритмы спецификаций.

Комплексный показатель сложности БА определяется следующим образом: $S_a = S^{спр}{}_a + S^{шап}{}_a + S^{спец}{}_a$. (3.9)

Для качественной оценки предложенной модели на основе объектно-ориентированного подхода необходимо ввести коэффициент насыщенности классового дерева k. Этот коэффициент характеризует количество потомков у родителя в дереве.

Предлагается определять исходные данные, необходимые для оценки сложности ОР-модели данных, используя следующие параметры: количество справочников $N_1' = N_1$, количество заголовков $N_2' = \frac{N_2 \cdot k - 1}{k - 1}$, количество спецификаций $N_3' = \frac{N_3 \cdot k - 1}{k - 1}$, количество атрибутов на справочниках $A_1' = A_1$, количество атрибутов на заголовках $A_2' = \frac{A_2}{U_2}$, где $U_2 = \frac{ln(N_2)}{ln(k)} + 1$ - количество

уровней в поддереве заголовков, количество атрибутов на спецификациях $A_3' = \frac{A_3}{U_3}$, где $U_3 = \frac{ln(N_3)}{ln(k)+1}$ - количество уровней в поддереве спецификаций, количество вторичных ключей на справочниках $S'_{fk1} = S_{fk1}$, количество вторичных ключей на заголовках $S'_{fk2} = \frac{S_{fk2}}{U_2}$, количество вторичных ключей на спецификациях $S'_{fk3} = \frac{S_{fk3}}{U_3}$, количество БА справочников $S'_{БА1} = S_{БА1}$, количество БА заголовков $S'_{БА2} = \frac{S_{БА2}}{U_2}$, количество БА спецификаций $S'_{БА3} = \frac{S_{БА3}}{U_3}$.

Для оценки сложности ОР-модели логической модели представления данных использованы следующие выражения:

Общая атрибутивная сложность: $A' = \sum_{i=1}^{3} A'_i \cdot N'_i$. (3.10)

Сложность связей в модели данных: $S'_{исх} = \sum_{i=1}^{3} S'_{fk\,i} \cdot N'_i$, (3.11)

$$S'_{вх} = N'_3 + \sum_{i=1}^{3} S'_{fk\,i} \cdot N'_i \,, \quad (3.12)$$

$$S'_{насл} = N'_1 + \sum_{i=2}^{3} \frac{S'_{fk\,i} \cdot k - 1}{k - 1}, \quad (3.13)$$

где $S'_{насл}$ – количество связей наследования, которые появляются в ОР-модели данных БД.

Общая сложность связей классово-ориентированной модели данных БД определяется как $S'_{св} = S'_{вх} + S'_{исх} + S'_{насл}$. (3.14)

Сложность алгоритмов для справочников определена как

$$S'^{спр}_{a} = S'_{БА1} + (\sum_{i=1}^{2} k_i \cdot A'_i) + k_3 \cdot \frac{S'_{исх}}{N'_1} . \quad (3.15)$$

Сложность алгоритмов для заголовков можно определить с помощью следующей зависимости $S'^{шап}_{a} = S'_{БА2} + k_1 \cdot A'_2 + k_2 \cdot S'_{fk2} + k_3 \cdot \frac{N'_3}{N'_2}$. (3.16)

Сложность алгоритмов для спецификаций определяется как

$$S'^{спец}_{a} = S'_{БА3} + k_1 \cdot A'_3 + k_2 \cdot S'_{fk3} . \quad (3.17)$$

Комплексный показатель сложности БА ОРБД:

$$S'_{а} = S'^{спр}_{а} + S'^{шап}_{а} + S'^{спец}_{а} . \quad (3.18)$$

Интегральная оценка сложности классической реляционной модели данных выражается следующей зависимостью $S = A\,a_1 + S_{св}\,a_2 + S_а\,a_3$. (3.19)

Интегральная оценка сложности классово-ориентированной модели данных определяется с помощью следующей зависимости

$$S' = A'\,a_1 + S'_{св}\,a_2 + S'_а\,a_3 . \quad (3.20)$$

В соответствии с ГОСТ Р ИСО/МЭК 9126, для оценки качества БД предложен критерий надежности, как функция тестируемости

$$E_т = (Q_{тм}) / (Q_{ом}), \qquad (3.21)$$

где $Q_{тм}$ - количество элементов, отработавших в процессе тестирования и отладки, $Q_{ом}$ - общему числу элементов БД. Количество элементов БД является функцией сложности БД.

В ГОСТ Р ИСО/МЭК 9126 установлены две основные субхарактеристики оценки эффективности программных средств, это эффективность ресурсная и эффективность временная.

В качестве меры ресурсной эффективности автором предложено использовать меру объема индексов базы данных, так как и в РБД и ОРБД объем данных одинаковый, он различается количеством составных элементов. Установлено, что ресурсная эффективность БД может определяться как $E_р = f(V_{дан}, V_{доп.надст})$, где $V_{доп.надст} = f(V_{инд.})$.

Соответственно для РБД объем индексов определяется следующим образом

$$V_{инд.}^{РБД} = W_i * W_s * \frac{k_d^{\frac{ln(N)}{ln(k_d)}+1} - 1}{k_d - 1}. \qquad (3.22)$$

Для БД, в основе которой используется логическая ОР-модель представления данных, объем индексов определяется как:

$$V_{инд.}^{ОРБД} = \sum_{i=0}^{u} \frac{\frac{V_{1ш}}{\left(\frac{ln(Ws)}{ln(k)}+1\right)} * \frac{W_s}{k^i}}{k_d - 1} * \left(k_d^{\frac{ln(N*k^i)}{ln(k_d)}+1} - 1 \right), \qquad (3.23)$$

где $W_i = W_a \cdot k_i$ – количество индексов на таблице (количество атрибутов, коэффициент индексирования), W_s – количество сущностей, k_d – коэффициент насыщенности дерева *B+*, k – коэффициент насыщенности дерева ОРБД, N – количество записей в таблице.

Предложено использовать в качестве меры временной эффективности БД время выполнения 1й БТ: $E_{Бт} = f(T_{бт})$ – время выполнения бизнес транзакции, где $T_{бт} = k_1 \cdot T_s + k_2 \cdot T_i + k_3 \cdot T_u + k_4 \cdot T_d$, $k_1 >> k_2 >> k_3 >> k_4$.

В соответствии с этим время выполнения 1й БТ в РБД можно определяться по выражению

$$T_{бт} = f(T_{п.рбд}) = f(U_{п.рбд}) = c \cdot \left(\frac{ln(N)}{ln(k_d)} + 1 \right). \qquad (3.24)$$

Мера временной эффективности ОРБД определяются по следующей формуле:

$$U_{п.кобд} = \frac{ln(N*k)}{ln(k_d)} + 1, \qquad (3.25)$$

где U – количество уровней в дереве поиска, c – коэффициент вовлеченности в поиск, где $c = 1, .., k$ для систем оперативного учета $c \to k$.

Таким образом, можно сделать вывод, что при поиске по группе сущностей глубина поиска при использовании логической ОР-модели представления данных останется не зависящей от количества вовлеченных в поиск сущностей.

В заключении главы отмечаются, наблюдающееся снижение декларативной сложности при увеличении количества сущностей в системе, снижение количества связей (функциональных) при увеличении числа сущностей, при анализе структуры ОРБД. Делается заключение, что наилучшим значением коэффициента насыщенности классового дерева является «3», а при увеличении насыщенности классового дерева общая сложность системы будет стремиться к сложности исходной РБД.

В четвертой главе оценивается реализация предложенного метода структурной декомпозиции реляционных сущностей. В качестве основы была использована реляционная модель данных, разработанная и внедренная на предприятии ТРК «Июнь» г. Санкт - Петербург.

При экспериментальной апробации разработанной методики использовалась рабочая база данных, реализованная на базе СУБД MS SQL 2005. Основные характеристики: размер БД – 998568 Мб, количество таблиц БД – 283, количество атрибутов таблиц БД– 3176, количество связей – 702, количество триггеров – 484, интенсивность продаж – 20000 – 40000 прод./сут. (в среднем 120 зап. в БД./сек.). Были поставлены задачи:

1) Разработать логическую ОР-модель представления данных на основе существующей классической ER-модели базы данных.

2) Разработать ОР-модель хранения алгоритмов обработки данных, которая позволит упорядочить их хранение и обеспечить управление их вызовами.

3) Оценить качество полученной ОР-модели, используя критерии предложенные в работе.

Для оценки эффективности выполнено сравнение результатов работы исходной БД и с БД полученной после преобразования (см. таблицу 4.1)

Таблица 4.1 Сравнение результатов преобразования РБД в ОРБД

Характеристика	РБД	ОРБД
Количество таблиц (N)	283	399
Связи ($S_{св}$)	702	584
Атрибуты (A)	3176	1865
Триггеры ($S_{БА}$)	484	361

На основе полученных результатов выполнено качественная оценка сложности исходной базы данных и базы данных после ее преобразования.

Применение предложенной ОР-модели позволило снизить затраты времени на обработку одного элемента по сравнению с исходной на 30%.

Следовательно, можно сделать заключение о том, что предложенный метод построения логической ОР-модели представления данных позволил повысить производительность БД, а также решить задачу по снижению ее сложности, так как сложность исходной РБД $S = 1357$, а полученная ОРБД $S' = 879$.

Делается вывод о том, что применение метода построения логической ОР-модели представления данных позволяет снизить сложность ОРБД на 34% по отношению к исходной РБД.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В процессе выполнения работы получены следующие основные результаты.

1. Разработана интегрированная объектно-реляционная логическая модель представления данных для больших реляционных БД и метод преобразования классической реляционной логической модели представления данных в предложенную ОР-модель. Применение ОР-модели позволило снизить структурную избыточность логической модели представления данных, а также предотвратить появление структурных аномалий в условиях динамического изменения как схемы данных, так и самих данных.

2. Разработана объектно-реляционная модель хранения алгоритмов обработки данных, а также метод грануляции реляционных алгоритмов классической реляционной БД в ОР-модель. Применение ОР-модели хранения алгоритмов позволило снизить семантическую избыточность реляционных алгоритмов обработки данных и упорядочить алгоритмы в виде синтезированной древовидной структуры их хранения и, таким образом, обеспечить эффективное управление обработкой данных.

3. Разработаны критерии качественной оценки БД потоковой обработки информации, в основе которых используется комплексный критерий сложности логических моделей представления данных. Разработанные критерии позволяют качественно оценить логические модели представления данных, определить их основные характеристики.

4. Проведено исследование эффективности предложенной ОР-модели логического представления данных и алгоритмов в рамках системы потоковой обработки складских документов крупной торгово-розничной сети ТРК Петербург г. Санкт-Петербург. Применение предложенной логической ОР-модели представления данных позволило снизить сложность БД на 36% по сравнению с изначально реализованной БД, а также повысить надежность работы БД и эффективность ее модернизации в процессе жизненного цикла.

ОСНОВНЫЕ ПОЛОЖЕНИЯ ДИССЕРТАЦИИ ОПУБЛИКОВАНЫ В РАБОТАХ

В рецензируемом журнале из списка ВАК

1. Проблемы организации структуры данных в сверхбольших базах данных / Масленников В.А., А.А. Левков // Научно-технический журнал “Системы управления и информационные технологии”, 2007, № 3.1(29), С. 169-176.

Монография

2. Структурная оптимизация многомерных систем хранения данных / Ю.С. Кабальнов, А. А. Левков, В. А. Масленников // М.: УГАТУ, 2007. – 213 с.

В других изданиях

3. Использование многомерных технологий хранения и обработки информации при создании базы данных “Расписание учебных занятий” / Масленников В.А., П.А. Алкаев, Ю.С. Кабальнов, А.Л. Калинина, Г.Ф. Низамова // Мавлютовские чтения. Матер. всерос. науч.-техн. конф. Уфа, УГАТУ, 2006, Т.1, С. 9-14.

4. Повышение эффективности реинжиниринга сложных БД при помощи использования объектно-ориентированных БД / Масленников В.А., А.А. Левков // Проблемы техники и технологий телекоммуникаций. Матер. VII междунар. науч. конф., Самара, 2006, С. 34-36.

5. Реализация наследования при оптимизация объектно-ориентированной модели данных / Масленников В.А., А.А. Левков // Проблемы Техники и Технологий Телекоммуникаций. Матер. VII междунар. науч. конф., Самара, 2006, С. 48-50.

6. Методология построения системы оперативного учета и документооборота / Масленников В.А., А.А. Левков // Мавлютовские чтения. Матер. всерос. молодеж. науч.-техн. конф., Уфа, УГАТУ, 2007, С. 41-43.

7. Способы оптимизации построения логических моделей данных ОО БД / Масленников В.А., А.А. Левков // Проблемы техники и технологии телекоммуникаций. Матер. VIII междунар. науч.-техн. конф., Уфа, УГАТУ, 2007, С. 144-146.

8. Методы оптимизации хранения информации об испытаниях авиационных двигателей / Масленников В.А., А.А. Левков // Информационные технологии и математическое моделирование. Матер. VI междунар. науч.-техн. конф., Томск, 2007, С. 55-58.

9. Оптимизация структуры базы данных оперативного учета и документооборота / Масленников В.А., А.А. Левков // Информационно-математические технологии в экономике, технике и образовании. Матер. II междунар. науч.-техн. конф., Екатеринбург, УГТУ-УПИ, 2007, С. 54-55.

10. Использование классовых алгоритмов и структур в реляционных базах данных / Масленников В.А., А.А. Левков // Решетневские чтения. Матер. XI междунар. науч. конф., Красноярск, СГАУ, 2007, С. 36-37.

11. Объектно-ориентированные модели для хранения данных / Маленников В.А. // Компьютерные науки и информационные технологии. Матер. IX междун. семин., Уфа, УГАТУ, 2007, С. 87-89.

Диссертант _____В.А. Масленников