

На правах рукописи

**Мещеряков Сергей Владимирович**

**ЭФФЕКТИВНЫЕ МЕТОДЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ  
ДЛЯ ЗАДАЧ УПРАВЛЕНИЯ СЕРВИСНЫМИ  
ПРОИЗВОДСТВЕННЫМИ СИСТЕМАМИ**

Специальности 05.13.06 – Автоматизация и управление технологическими процессами и производствами (сфера услуг)  
05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

Автореферат  
диссертации на соискание ученой степени  
доктора технических наук

Санкт-Петербург – 2007

Работа выполнена в Санкт-Петербургском государственном политехническом университете.

Научный консультант:

доктор физико-математических наук,  
профессор

Иванов В. М.

Официальные оппоненты:

доктор технических наук, профессор

Уткин Л.В.

доктор технических наук, профессор

Сироткин Я.А.

доктор технических наук, с.н.с.

Коршунов Г.И.

Ведущая организация:

ОАО «Машиностроительный завод «Арсенал»»  
(г. Санкт-Петербург)

Защита состоится “ 23 ” октября \_\_\_\_\_ 2007 г. в \_\_\_\_ часов на заседании диссертационного совета Д212.229.21 Санкт-Петербургского государственного политехнического университета по адресу:  
195251, Санкт-Петербург, ул. Политехническая, 29, Главное здание, ауд. 118.

С диссертацией можно ознакомиться в фундаментальной библиотеке Санкт-Петербургского государственного политехнического университета.

Автореферат разослан “    ” \_\_\_\_\_ 200   г.

Ученый секретарь  
диссертационного совета

Редько С. Г.

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность темы.** Перспективы развития современного общества тесно связаны с глобальным использованием информации. Национальные информационные ресурсы являются макроэкономической ценностью, потенциальным источником общественного богатства. Информация как носитель знания стала стратегическим фактором, который воздействует на структуру всей экономики и общественной жизни и в конечном счете определяет темпы развития всего общества.

Информатизация современных предприятий представляет собой экспоненциально нарастающее производство, переработку и использование огромных массивов информации во всех областях применения.

Наряду с ускоренным развитием компьютерной техники, информационные технологии (ИТ) в области создания информационных систем (ИС) и приложений баз данных (БД) совершили за последние годы гигантский скачок вперед. В XXI веке, после решения «проблемы 2000 года» и обновления парка компьютерного оборудования, создание информационных систем и технологий и их внедрение на предприятиях приобрело черты целой ИТ-индустрии с присущими ей особенностями:

- интеллектуальная специфика программной продукции, ее виртуальность (за исключением компьютерного оборудования);
- высокотехнологичные средства производства, позволяющие быстро и эффективно создавать качественные программные продукты в соответствии с растущими потребностями предприятий;
- инновационность отрасли, непрерывное развитие и усовершенствование всех составляющих ИТ-сферы – компьютерного оборудования, телекоммуникационных систем, программного обеспечения (ПО), информационных услуг.

Потребность предприятий в разработке новых эффективных методов проектирования ИС и приложений БД обусловлена многими факторами:

- высокими темпами развития индустрии информационных технологий;
- увеличением потребительского спроса на программную продукцию;
- большим разнообразием видов ПО, которые так или иначе связаны с построением ИС и приложений БД;
- увеличением объемов обрабатываемой информации на предприятии, необходимостью ее структуризации и интеграции в единую систему управления и принятия решений;
- информатизацией всех уровней современной экономики, включая малые и средние предприятия;
- государственной политикой в области информатизации, ИТ-стандартов и информационной безопасности.

Помимо увеличения объемов информации, усложняются и методы ее обработки. Совершенствуются технологии создания ИС и приложений БД на фоне появления новых программно-инструментальных средств и систем управления базами данных (СУБД). Для создания крупных ИС привлекаются трудозатраты в сотни человеко-лет и современная технологическая база, задействованы высококвалифицированные специалисты таких профессий, как системные аналитики, менеджеры проектов, программисты, тестировщики ПО, администраторы БД, системные интеграторы.

Объективная потребность контролировать процесс создания сложных ИС, прогнозировать и гарантировать стоимость разработки, сроки и качество результатов привела к необходимости перехода от кустарных способов к индустриальным технологиям и инженерным методам и средствам создания ПО. Поэтому на каждом из этапов создания ИС актуально применение эффективных методов и приемов, позволяющих снизить издержки, повысить продуктивность процесса разработки ПО, улучшить качество производимой ИТ-продукции и предоставляемых ИТ-услуг.

Решению различных теоретических и практических проблем построения ИС в технических системах посвятили свои работы ведущие ученые и специалисты А.М. Вендров, А.Ю. Грачев, М. Гутман, Н. Елманова, Д.П. Зегжда, П.Д. Зегжда, Ю.А. Зеленков, Д. Зенкин, В.Х. Инмон, Л. Калиниченко, Г.Н. Калянов, А. Коберн, Е.Ф. Кодд, С.Б. Кодд, Л. Козленко, С.Д. Кузнецов, А. Лукутин, И. Лукьяненко, Ю.А. Подколзин, Дж. Рой, Дж.М. Смит, Д.К. Смит, М. Стоунбрейкер, Д.Т. Чанг, П.П.-Ш. Чен, С. Чаудхари и др.

**Целью диссертационной работы** является разработка методологических и теоретических основ проектирования автоматизированных информационных систем и приложений баз данных в сервисных производственных системах.

Для достижения поставленной цели в работе решались следующие основные задачи:

- исследование и разработка эффективных методов проектирования баз данных с возможностью интеграции в единую систему сбора и обработки информации на предприятии;
- разработка концепции и построение объектно-реляционных моделей технических объектов с произвольным набором атрибутов;
- разработка новых методов проектирования иерархических структур данных, отличающихся большей производительностью, и их сравнительная оценка на основе выбранных критериев качества;
- апробация предложенных методов проектирования БД, разработка информационно-управляющей БД производственных объектов на предприятии;
- разработка эффективных методов переноса накопленной информации в новые структуры БД на основе анализа проблем интеграции и совместимости данных;

- исследование возможности и разработка концепции произвольных запросов к БД и их применение для формирования отчетов;
- разработка новых технологий представления в БД графической и другой информации нестандартного типа, ее параметризация и использование для автоматизации документооборота;
- анализ источников угрозы и разработка новых схем обеспечения защиты информации в БД, отличающихся большей надежностью.

**Объектом исследования** являются информационные системы и приложения баз данных в сервисных производственных системах, функционирующие в архитектуре «клиент-сервер».

**Предметом исследования** являются методы автоматизации проектирования баз данных и их приложений в производственных системах на всех этапах «жизненного цикла» программного продукта – от анализа и проектирования до внедрения и сопровождения.

**Методы исследования.** Для решения поставленных задач в работе использовались методы объектно-ориентированного проектирования и программирования, аппарат реляционной алгебры, теория множеств, методы оптимизации и математической статистики.

**Научная новизна** диссертационной работы заключается в следующем:

1. Созданы методологические и теоретические основы проектирования баз данных с интеграцией в единую систему сбора и обработки информации на предприятии. Предложены новые схемы представления данных и организации единого информационного пространства в целях интеграции различных служб предприятия в единую систему управления и принятия решений.

2. Разработана концепция создания информационной модели для задач управления техническим обслуживанием объектов грузоподъемного оборудования, которая является обобщением существующих моделей и служит основой для проектирования других более сложных иерархических моделей объектов производственных систем.

3. Впервые разработана универсальная структура данных для хранения и обработки информации об объектах производства, которая принципиально отличается от других решений тем, что позволяет неограниченно увеличить количество разнородных параметров, характеризующих техническое состояние объекта, и повысить эффективность обслуживания производства.

4. Предложены новые иерархические модели, не содержащие рекурсию, преимущество которых по критерию производительности по сравнению с известными подтверждено результатами вычислительного эксперимента.

5. Предложена новая формализованная методика конструирования информационно-управляющих моделей производственных объектов любой степени сложности для решения задачи единого информационного учета всего парка технологического оборудования, эксплуатируемого на предприятии.

6. Разработаны эффективные алгоритмы и типовые программные модули синхронизации данных между различными подсистемами СУБД, позволяющие исключить потери накопленной информации при ее преобразовании и переносе в новые управляющие структуры.

7. Предложены новые нетрадиционные методы формирования отчетной документации на основе анализа потребностей современных предприятий в автоматизации документооборота, которые отличаются от известных методов наличием обратной связи с приложением БД, гибкостью изменения старых и создания новых отчетов по результатам произвольных запросов к БД.

8. Сформулированы принципы и условия эффективного управления сервисными ресурсами. Предложены новые методы хранения и обработки нестандартных типов данных, которые невозможно реализовать встроенными средствами СУБД. Поставлена и решена задача параметризации технических чертежей производственных объектов и управления графическими построениями путем установления связи между объектами чертежа и их числовыми параметрами, хранимыми в БД.

9. Предложены новые нетрадиционные средства защиты информации от несанкционированного доступа (НСД) и алгоритм их функционирования, отличающиеся нестандартным подходом и большей надежностью для обеспечения информационной безопасности.

#### **Положения, выносимые на защиту:**

1. Методология проектирования иерархических, объектно-реляционных и многомерных структур данных.

2. Новая информационная модель описания производственных объектов, позволяющая неограниченно увеличивать количество атрибутов и добавлять описания новых классов оборудования без изменения структуры данных.

3. Новые безрекурсивные методы организации иерархии данных и результаты вычислительного эксперимента по сравнительной оценке их эффективности на основе известных критериев качества.

4. Методика конструирования информационных моделей производственных объектов произвольной сложности и разработанная на предприятии база данных автоматизированной системы технической подготовки производства (АСТПП).

5. Эффективные алгоритмы преобразования информации, накопленной в устаревших системах, и ее переноса с минимизацией ошибок в разработанные управляющие структуры.

6. Новые методы автоматизации документооборота посредством произвольных запросов к БД, параметризации технических чертежей и управления графическими построениями.

7. Новые средства защиты информации в БД, обеспечивающие более высокий уровень информационной безопасности.

### **Практическая ценность и реализация основных результатов работы.**

В основу диссертации положены научные результаты, полученные лично автором при непосредственной разработке прикладных информационных систем и технологий для предприятий северо-западного региона России:

- база данных комплексного обследования крановых путей CraneRoad (1999-2005 г., ЗАО «РАТТЕ»);
- автоматизированное рабочее место обследователя грузоподъемных кранов Cranes (1997-2002 г., ЗАО «РАТТЕ»);
- информационные системы документооборота и платежных операций DeloPro (1998-2000 г.), Vouchers (2001-2005 г., ЗАО «РАТТЕ»);
- программно-аппаратный комплекс информационной защиты БД от НСД Security Removal Gangster Program on Hard Disk (2000-2001 г., ЗАО «АМН»);
- автоматизированные ИС учета состояния технологического оборудования – сосудов, аппаратов и клапанов SOS, грузоподъемных машин GPM, учета влияния коррозии COR (2000-2005 г., ООО «ПО КИНЕФ»).

Основные теоретические положения диссертации использованы в учебном процессе СПбГПУ:

- на кафедре транспортных и технологических систем механико-машиностроительного факультета по дисциплине «Основы автоматизированного проектирования» для студентов по направлению 551800 – «Технологические машины и оборудование», специальностям 170900 – «Подъемно-транспортные и строительные машины» и 220900 – «Автоматизация технологических производств» (2002 г.);
- на кафедре информационных машиностроительных технологий механико-машиностроительного факультета по дисциплинам «Машиностроительная информатика» и «Технические средства САПР» в рамках инженерной и магистерской подготовки (1999 г.);
- на кафедре информатики Института международных образовательных программ при постановке нового курса «Базы данных» для студентов специальности 080801 – «Прикладная информатика в дизайне» (2003 г.).

Эффективность реализованных методов, алгоритмов и программного обеспечения подтверждена 4 актами о внедрении.

**Апробация работы.** Основные результаты научных разработок, выполненных автором по теме диссертации, представлены на международных и российских конференциях: «Совершенствование учебного процесса при подготовке инженеров-механиков» (СПбГТУ, 1994 г.), «Вакуумная наука и техника» (Гурзуф, 1994 г.), «Инновационные наукоемкие технологии для России» (СПбГТУ, 1995 г.), «Прогрессивные конструкции и технологии в машиностроении» (СПбГТУ, 1996 г.), «Фундаментальные исследования в технических университетах» (СПбГПУ, 1997, 2002, 2003 г.г.), «Современные научные школы: перспективы развития» (СПбГТУ, 1998 г.), «XXVII неделя науки СПбГТУ»

(СПбГТУ, 1998 г.), «XXIX неделя науки СПбГТУ» (СПбГТУ, 2000 г.), «Высокие интеллектуальные технологии образования и науки» (СПбГПУ, 2002 г.), «Оптимизация транспортных машин» (Ульяновск, УлГТУ, 2003 г.), «Научно-технические проблемы прогнозирования надежности и долговечности конструкций и методы их решения» (СПбГПУ, 2003 г.), «Современные проблемы информатизации в системах моделирования, программирования и телекоммуникациях» (Воронеж, ВорГТУ, 2004 г.), «Формирование профессиональной культуры специалистов XXI века в техническом университете» (СПбГПУ, 2004 г.), «Информационные технологии моделирования и управления» (Воронеж, ВорГТУ, 2004 г.), The 30th International Conference for the Resource Management and Performance Evaluation of Enterprise Computing Systems (Las Vegas, USA, 2004), а также на научно-методических семинарах кафедр информатики и информационных машиностроительных технологий СПбГПУ (1999-2005 г.).

**Публикации.** Общее количество публикаций составляет 65 наименований. По теме диссертации опубликовано 37 научных трудов общим объемом 54 п. л., среди которых 1 монография, 11 статей (в т. ч. 4 из списка ВАК для докторских диссертаций), 14 тезисов докладов, 8 учебных пособий, 16 работ без соавторов.

**Структура и объем работы.** Диссертация включает перечень сокращений, введение, 7 глав, заключение, библиографию из 150 наименований и 8 приложений. Основное содержание работы изложено на 296 страницах текста, в т. ч. 19 таблиц, 72 рисунка.

## СОДЕРЖАНИЕ РАБОТЫ

**Во введении** обоснована актуальность проблемы повышения эффективности проектирования баз данных для отечественной индустрии информационных технологий. Сформулирована цель диссертационной работы, перечислены задачи исследований, научная новизна, апробация, практическая ценность и реализация основных результатов работы.

**В первой главе** приведены основные тенденции развития информатизации российской экономики в начале XXI века, проведен критический обзор состояния качества ИТ-продукции и уровня информационной безопасности.

Несмотря на отставание в развитии российской ИТ-индустрии от западных и некоторых восточных стран, ее устойчивый ежегодный рост сохраняется на уровне 30%. В структуре ИТ-отрасли выделяются четыре ведущих сегмента (рис. 1). Главной тенденцией является *увеличение доли информационных услуг*, которая может достигать половины затрат на автоматизацию. Предприятиям любого масштаба неэффективно наращивать собственные ИТ-службы, которые уже не могут отследить стремительный прогресс ИТ. Поэтому их функции передаются на аутсорсинг внешним организациям, где концентрируются ИТ-специалисты.



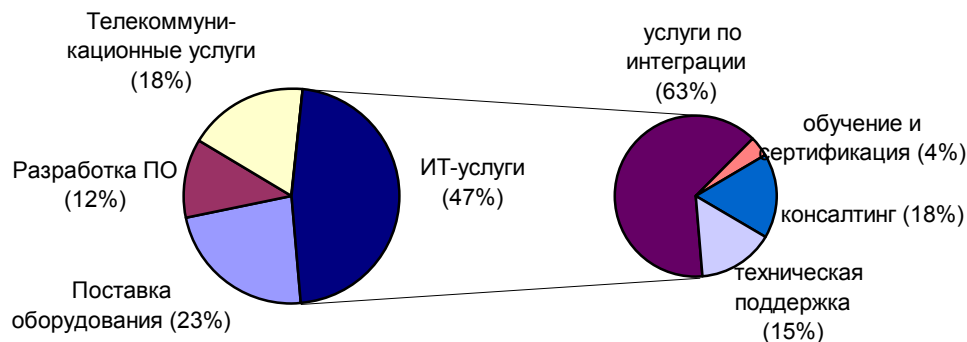


Рис. 1. Структура ИТ-отрасли российской экономики

Из всех форм взаимодействия предприятий с поставщиками ИТ-услуг чаще всего используется модель аутсорсинга отдельных задач (заказных проектов «под ключ»), реже аутсорсинг ресурсов (аренда специалистов), но наиболее актуальной и перспективной является *интеграция всех информационных ресурсов на предприятии*.

В отличие от западных концепций комплексной интеграции и систем принятия решений (например, SAP), большинство российских предприятий в первую очередь автоматизируют отдельные приоритетные процессы, чаще всего бухгалтерский учет, управление поставками и сбытом (так называемая «лоскутная» автоматизация). Большие затраты и необходимость кардинального изменения традиционно сложившихся информационных потоков на предприятии – главные препятствия для внедрения интеграционных решений.

Для успешного функционирования предприятий в современных условиях руководители внедряют стратегию управления и контроля качества продукции на уровне государственных и международных стандартов ISO 9000. Качество ИТ-продукции во многом зависит от грамотной организации тестирования и других методов контроля ПО, рассмотренных в главе 5.

С развитием сетевых технологий и единого мирового информационного пространства Интернет особую значимость приобрели болезненные вопросы информационной безопасности. Главной причиной небезопасности Интернет является открытость его архитектуры, основанной на модели «клиент-сервер», при которой именно серверы чаще всего подвержены хакерским атакам.

Состояние информационной безопасности на предприятиях характеризуется низким уровнем защищенности данных. Руководители стереотипно считают это технической проблемой ИТ-служб, хотя многие причины уязвимости заключаются во внутренней организации средств защиты, в злом умысле или компьютерной безграмотности сотрудников.

Большинство угроз утечки или потери информации исходит из внешних источников – информационный шум (спам), компьютерные вирусы, хакерские атаки и др. Согласно статистике, число вирусных инцидентов в мире ежегодно практически удваивается. Они распространяются молниеносно и способны полностью парализовать работу компьютерной сети предприятия. Общий ущерб от вирусного проникновения и ликвидации последствий может составлять миллиарды долларов. Информационная безопасность становится важнейшей составляющей национальной безопасности любого государства.

**Вторая глава** посвящена *методологии проектирования ИС* как формального процесса, дана *характеристика ИС как объекта исследования*. Приведена терминология программной инженерии – программный продукт (или услуга), жизненный цикл, технология и программно-инструментальные средства, технологическая операция, роль и др. Выделены технологические этапы производства ИС – анализ и постановка задачи, проектирование БД и создание прототипа, практическая реализация приложений БД, внедрение ИС и опытная эксплуатация, техническая поддержка и информационные услуги по сопровождению в течение всего «жизненного цикла» программного продукта (вплоть до выхода из употребления всеми пользователями).

Эффективность технологий создания ПО оценивается критериями:

- продуктивность (количество разработанных программных модулей, объем реализованных сервисных функций);
- качество создаваемого ПО (функциональная полнота, число дефектов и ошибок, обнаруженных в процессе эксплуатации);
- трудоемкость разработки ПО (количество человеко-дней);
- затраты на внедрение (суммарные расходы на разработку или приобретение ПО, обучение пользователей и услуги по сопровождению);
- эффект от использования ПО (положительный баланс между вложенными средствами, расходами до и после внедрения);
- срок окупаемости затрат на внедрение ПО (интервал времени от начала разработки ПО до момента полного возврата инвестиций).

Современные ИС реализуются в сетевой архитектуре «клиент-сервер» (одно-, двух-, трехуровневой или распределенной). Все операции взаимодействия прикладной программы пользователя (клиентской части) с данными на сервере – хранение, оперативный ввод данных, статистическая обработка, генерация отчетов, управление ресурсами – производятся посредством транзакций на языке запросов стандарта SQL. В зависимости от распределения функций и вычислительной нагрузки между клиентом и сервером, а также исходя из требований объекта автоматизации, реализуется та или иная концепция построения ИС (рис. 2). При этом наиболее важным критерием оптимизации производительности ИС является минимизация времени отклика (ожидания ответа) на возникающий поток (напор) запросов.

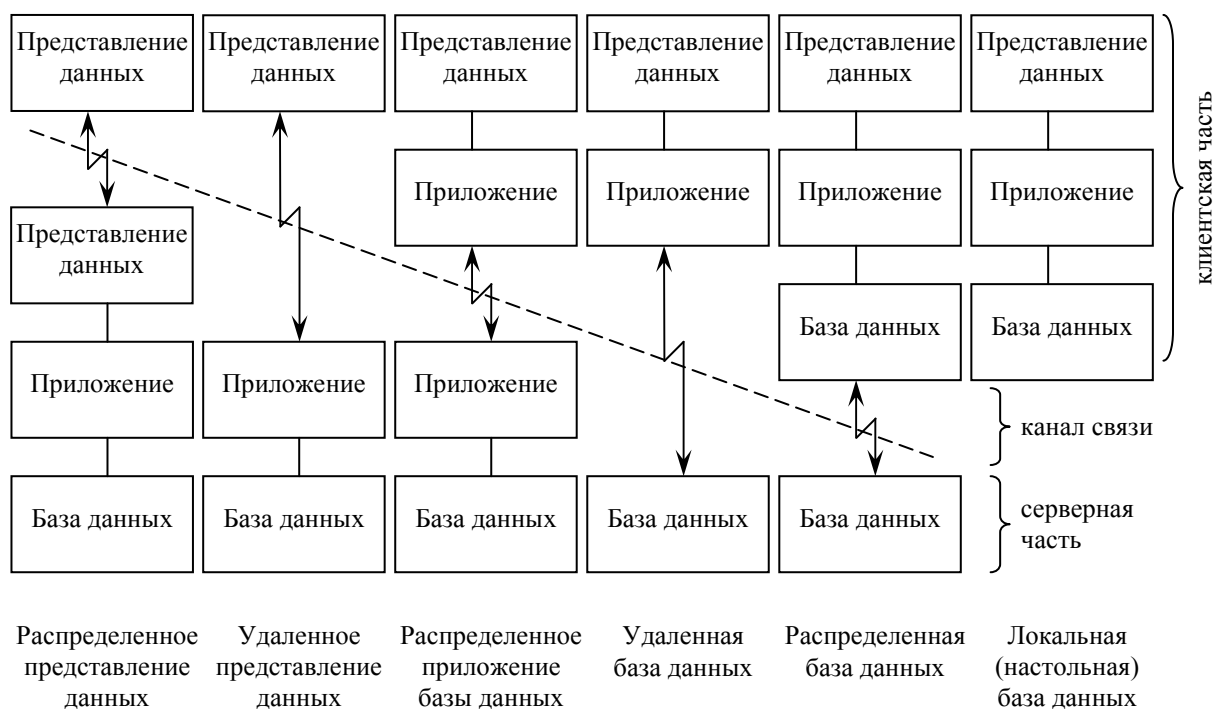


Рис. 2. Распределение функций между компонентами «клиент-сервер»

Классические модели обеспечения жизненного цикла ИС – каскадная, по спирали, поэтапная с промежуточным контролем, схема «водопада» – хотя и способствуют хорошему качеству ИС, но все же не пользуются популярностью у современных разработчиков из-за высокой трудоемкости и временных затрат. К сожалению, сроки практической реализации и внедрения ИС (табл. 1) часто имеют более важное значение, чем качество разработки.

*Проектирование БД является наиболее трудоемким этапом построения ИС* и заключается в последовательном решении следующих задач:

- разработка концептуальной модели БД (диаграммы потоков данных, «сущность-связь», переходов состояний, сети Петри);
- построение логической модели (структуры БД);
- выбор аппаратной конфигурации и программных средств;
- создание физической модели БД (для выбранной СУБД);
- разработка приложений – автоматизированных рабочих мест (АРМ).

Таблица 1. Средние сроки реализации основных этапов разработки ИС

№ этапа	Содержание этапов разработки ИС	Сроки реализации, мес.
1	Анализ и проектирование системы	1-3
2	Программирование и отладка функционального ядра	2-4
3	Тестирование и опытная эксплуатация	1-4
4	Доработка и сопровождение проекта	3-6

В задачах проектирования БД используются методы структурного анализа, в основе которых лежат алгоритмы расчленения больших систем на компоненты (декомпозиция) и их иерархическая организация. При решении задачи планирования ремонтов оборудования применен метод декомпозиции отношения «Дефектная ведомость – объект ремонта». Введено понятие новой сущности «Ремонтные мероприятия» (рис. 3). «Дефектная ведомость» получена путем реляционной операции выборки кортежей по отношению к заданному «Объекту ремонта» и по условию одинаковости атрибута «Дата» с последующей реляционной операцией проецирования результата выборки на необходимые атрибуты «Ремонтных мероприятий», как показано ниже.

- $R_1$  = «Акты расследования неполадок» («Номер», «Дата», «Причина», «Признак остановки»);  
 $R_2$  = «Акты обследования» («Номер», «Дата», «Вид ремонта», «Текущие мероприятия»);  
 $R_3$  = «Акты технического состояния после ремонта» («Номер», «Дата», «Исполнитель», «Проведенные мероприятия», «Перспективные мероприятия»);  
 $D$  = «Дефектная ведомость на ремонт» («Номер», «Дата», «Название», «Объект ремонта»).

После реляционной операции соединения

$$A = R_1 \cup R_2 \cup R_3 = \{a: a \in R_1 \vee a \in R_2 \vee a \in R_3\}$$

получены новые отношения

- $O$  = «Объект ремонта» («Номер», «Название», «Месторасположение»),  
 $A$  = «Ремонтные мероприятия» («Номер», «Дата», «Вид ремонта», «Причина, мероприятия», «Исполнитель»)

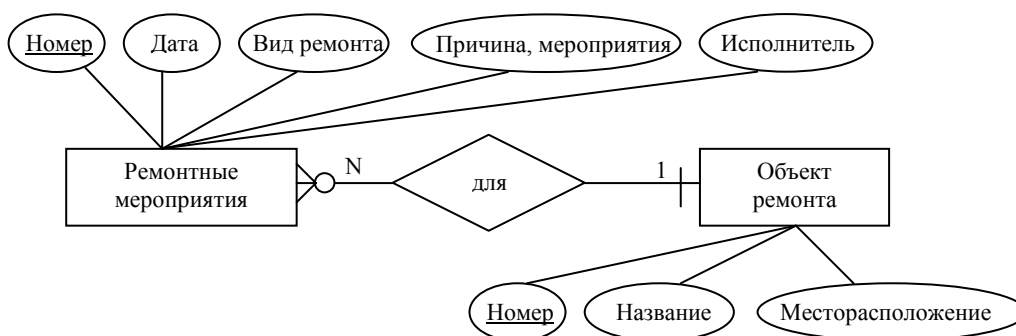


Рис. 3. Диаграмма «сущность-связь» после декомпозиции

с набором атрибутов  $a_1 = \text{«Номер»}$ ;  $a_2 = \text{«Дата»}$ ;  $a_3 = \text{«Вид ремонта»}$ ; ...  
Реляционная операция выборки подмножества кортежей, удовлетворяющих условиям  $a_2 = \text{«Дата»}$ ;  $O = \text{«Объект ремонта»}$ , будет иметь вид:

$$S = D [A \theta O, A \theta a_2] = \{d: d \in D \wedge (d[A] \theta O) \wedge (d[A] \theta a_2)\}.$$

В результате реляционной операции проецирования результата выборки  $S$  на атрибуты  $a_1, a_2, a_3, \dots$  получим:

$$P = S [a_1, a_2, a_3, \dots] = \{s[a_1, a_2, a_3, \dots]: s \in S\}.$$

Запрос на языке реляционного исчисления может быть записан как

Выбрать «Объект ремонта, вид ремонта, причина, мероприятия, исполнитель» из «Ремонтные мероприятия» такие, что существует «Объект ремонта» для заданной «Дата».

Реляционная операция взятия проекции  $P$  не приведет к возникновению кортежей-дубликатов в вертикальном срезе данных, т. к. кардинальность связи «Ремонтные мероприятия – объект ремонта» (многие к одному, обязательная на стороне сущности «Объект ремонта») обеспечивает уникальность объектов ремонта с атрибутами «вид ремонта, причина, мероприятия, исполнитель».

**Третья глава** посвящена *методам разработки оптимальных структур БД* на основе главных критериев оптимального размещения данных – производительность, расход ресурсов внешней памяти, надежность и простота администрирования.

Оптимальное решение всегда представляет собой компромисс между характеристиками БД, противоречащими друг другу. Так чтобы получить выигрыш в скорости обработки данных, приходится дополнительно тратить внешнюю память. В работе даны рекомендации по оптимизации БД:

- централизованное внесение изменений в структуры данных;
  - выявление и удаление неиспользуемых объектов БД (таблиц, процедур);
  - анализ использования объектов БД в приложениях;
- а также сформулированы мероприятия, направленные на рациональное распределение и эффективное управление информационными ресурсами, при выполнении которых можно достичь увеличения производительности до 400%:
- установка оптимальных размеров блоков данных;
  - уменьшение фрагментации дискового пространства;
  - группировка хранимых данных по логическим разделам БД;
  - децентрализация, распределение БД по разным физическим устройствам;
  - оптимизация хранения больших двоичных объектов (BLOB/CLOB);
  - индексирование таблиц данных и поддержание статистики БД.

Современные СУБД объектно-реляционного типа обладают развитыми средствами для эффективного проектирования БД, за исключением механизмов организации иерархии объектов с логическими связями «предок-потомок», образующими древовидную структуру. Любой элемент иерархии может быть одновременно потомком для одного и предком для других компонентов. Классическим способом реализации многоуровневой структуры реляционных таблиц служит следующее описание на языке SQL:

```
create table tbl_name (  
    id integer not null primary key,  
    name varchar (254) not null,  
    parent_id integer references tbl_name (id) constraint fk_parent_id,  
);
```

Главная проблема ссылочных связей ( $parent\_id \rightarrow id$ ) заключается в наличии рекурсивности реляционных отношений, а рекурсия в запросах значительно ухудшает производительность СУБД и может привести к ее зависанию, т. е. отказу в обслуживании, требующему вмешательства администратора БД, что крайне недопустимо для клиент-серверных систем.

Для решения проблемы рекурсивности использована предпосылка, что на практике иерархия обычно не бывает бесконечной и имеет ряд ограничений:

- конечная глубина вложенности, например структура подразделений любого предприятия всегда ограничена по уровням – цех, участок, рабочее место;
- максимальное количество непосредственных потомков одного предка, равно как в любой бригаде число рабочих конечно;
- ограничение на общее количество потомков одного уровня иерархии аналогично тому, что количество цехов завода строго фиксировано.

Граничные условия устанавливаются в каждом конкретном случае практической реализации. С учетом этих ограничений и на основе анализа всех типов рекурсивных связей в работе предложены *новые методы построения иерархии данных и процедуры их обработки, не содержащие рекурсию*:

- хранение уровня иерархии;
- иерархия с неограниченной глубиной и конечным числом потомков;
- потабличное хранение конечного числа уровней вложенности;
- иерархия с поразрядным (левым или правым) ключом кодирования;
- древовидная структура с хранением границ ветви;
- иерархия с компонентом типа «узел»;

а также их сравнительная характеристика по следующим критериям универсальности и эффективности:

- потребность в модификации таблиц при изменении уровней вложенности;
- избыточность атрибутов, наличие незаполненных полей таблицы;

- неограниченная глубина (количество уровней) иерархии;
- произвольное число потомков одного уровня вложенности;
- рекурсивность запросов при получении полного пути до корня дерева;
- быстрое получение всех элементов одного уровня иерархии;
- легкость получения всех (в т. ч. непрямых) потомков произвольного предка;
- контроль правильности ссылки на родителя без сложных вычислений;
- перемещение ветвей дерева стандартными средствами SQL;
- поддержка в компонентах известных систем программирования.

Результаты сравнительных тестов производительности различных вариантов иерархических структур представлены в главе 5.

В отличие от иерархии объектов, иерархия атрибутов может быть организована встроенными средствами объектно-реляционных СУБД на основе принципа наследования. Но в практике проектирования сложных технических систем представление объектов различных классов может насчитывать сотни и даже тысячи параметров. *Реализация широких таблиц значительно снижает производительность БД.* Разбиение широкой таблицы на несколько узких принципиально не решает проблему быстродействия, так как при обращении к данным потребуется операция соединения (join). Индексирование для широких таблиц тоже неэффективно, поскольку быстродействие поиска по индексу экспоненциально зависит от длины таблицы, а зависимость между скоростью последовательного перебора и размером выборки пропорциональная:

$$\begin{aligned} \exp(t_i) &\sim n, \text{ или } t_i \sim \ln(n), \\ t &\sim n \cdot k, \end{aligned}$$

где  $t$  – время последовательного перебора,  $t_i$  – время поиска с использованием индекса  $i$ ,  $n$  – длина таблицы (количество записей),  $k$  – ширина таблицы (количество атрибутов).

При решении проблемы широких таблиц каждый класс объектов абстрактно описан индивидуальным набором атрибутов. В то же время замечено, что некоторые атрибуты являются общими с другими классами, а с другой стороны, внутри класса может быть избыточность.

Пусть задана следующая схема реляционных отношений для объектов различных классов:

$$\begin{aligned} R_1(a_1, \dots, a_n, b_1, \dots, b_m) &= R_1[(a \parallel b): a \in A \wedge b \in B], \\ \dots \\ R_p(a_1, \dots, a_n, c_1, \dots, c_k) &= R_p[(a \parallel c): a \in A \wedge c \in C], \end{aligned}$$

где  $A = \{a_1 \dots a_n\}$  – набор атрибутов, характеризующих объекты всех классов;  $B = \{b_1 \dots b_m\}$  – набор индивидуальных атрибутов для объектов 1-го класса;  $C = \{c_1 \dots c_k\}$  – набор индивидуальных атрибутов для объектов  $p$ -го класса.

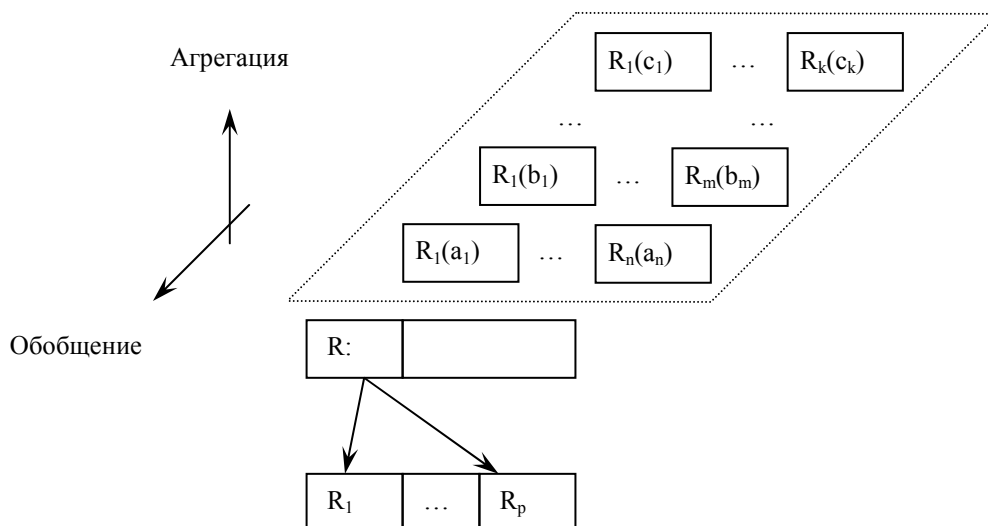


Рис. 4. Графическая модель обобщения реляционных отношений

Графическая нотация этой модели является трехмерной (рис. 4). Агрегация и обобщение представляются ортогонально – агрегация в плоскости листа, а обобщение в плоскости, перпендикулярной листу. Агрегированные объекты более высокого уровня иерархии показаны вверху, а более низких уровней – внизу модели. Атрибуты родовых объектов верхнего уровня расположены на поверхности листа, а более низких уровней – сзади (в глубине) листа.

Каждое из отношений  $R_{ij}$  служит примитивом структурирования для спецификации обобщений в реляционной модели и является либо идентификатором определенного типа, либо ключевым указателем на внешний домен. Такая объектно-реляционная модель специфицирует одновременно две абстракции – агрегацию связей между классами объектов от  $R_1$  до  $R_p$ ; обобщение атрибутов, характеризующих объекты  $R_{ij}$ .

*Унифицированный подход к обобщению реляционных отношений* заключается в том, что для исходных отношений  $R_1...R_p$  выполняется реляционная операция соединения (join), результатом которой является следующее отношение:

$$R_1(a_1, \dots, a_n, b_1, \dots, b_m, c_1, \dots, c_k) = \\ = R_1[(a \parallel b \parallel c): a \in A \wedge b \in B \wedge c \in C].$$

На рис. 5 показан пример обобщения реляционных отношений объектов грузоподъемного оборудования (ГПО) различных классов – грузоподъемных механизмов (ГПМ), лифтов, крановых путей. Для обобщения схожих параметров в дополнение к операции соединения потребуется выборка и объединение соответствующих атрибутов в реляционном отношении:



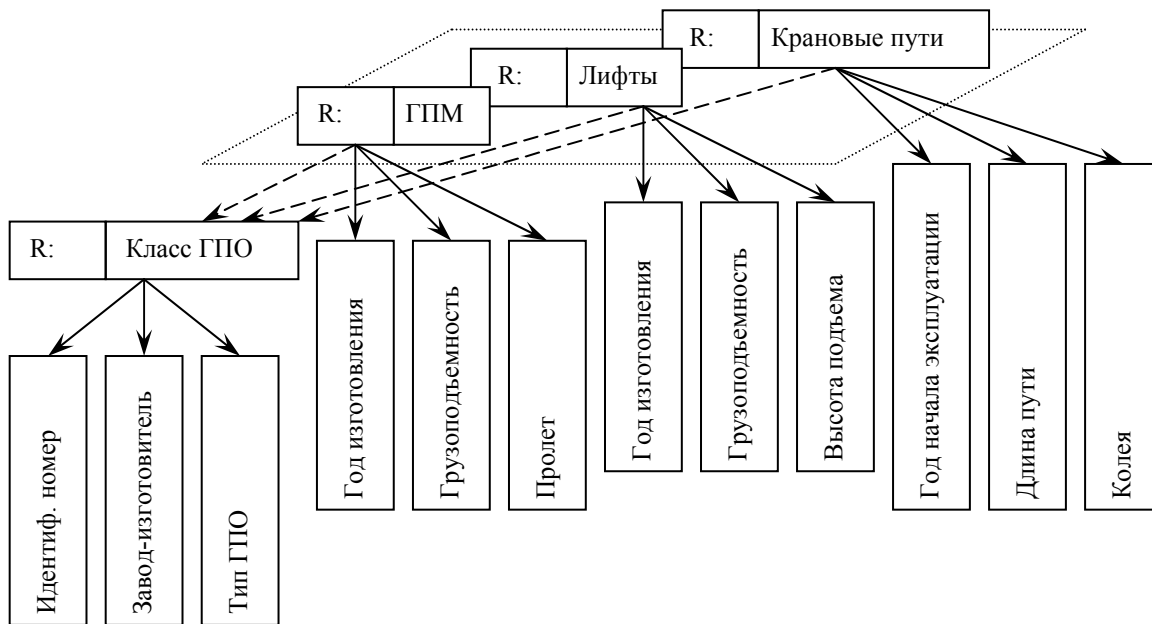


Рис. 5. Пример обобщения реляционных отношений объектов ГПМ

$$R_1(a_1, \dots, a_n, b_1, \dots, b_m, c_1, \dots, c_k) = \\ = R_1[(a \parallel b \parallel c): a \in A \wedge b \in B \wedge c \in C \wedge (B \theta C)].$$

Унифицированное обобщение позволяет сократить количество таблиц, но неизбежно приводит к увеличению ширины и появлению избыточности в результирующей таблице. Тем не менее, такой подход, основанный на совместном применении агрегации и обобщения, позволяет определить богатое множество моделей технических объектов.

Графическая модель (рис. 4) наглядно демонстрирует, что реляционные отношения  $R_{ij}$  позволяют путем их композиции описывать объекты различных классов и любой степени сложности. Такой процесс аналогичен конструированию больших сложных объектов из набора примитивов.

Метод конструирования информационной модели объектов БД основан на предварительном описании следующих реляционных отношений:

$$R_1(o_0, o_1, o_2), \\ R_2(p_0, p_1, p_2, p_3), \\ R_3(v_1, v_2, v_3), \\ R_4(c_0, c_1, c_2),$$

где  $R_1$  – описание объектов БД;  $R_2$  – описание учетных параметров;  $R_3$  – связь объектов и учетных параметров с их числовыми значениями;  $R_4$  – классифи-

кация объектов БД;  $o_0, p_0, c_0$  – ключевые атрибуты соответственно отношений  $R_1, R_2, R_4$ ;  $o_1$  – наименование объекта;  $o_2$  – принадлежность объекта классу;  $p_1$  – наименование параметра;  $p_2$  – тип значений параметра (числовой, строковый, дата/время, внешняя ссылка и т. д.);  $p_3$  – единица измерения параметра;  $v_1$  – ссылка на описание объекта ( $R_1$ );  $v_2$  – ссылка на описание параметра ( $R_2$ );  $v_3$  – значение параметра (или внешняя ссылка);  $c_1$  – наименование класса;  $c_2$  – классификационный признак (или ссылка на класс).

Главное преимущество такой революционной модели данных заключается в универсальности представления объектов различных классов с разнородными параметрами. Заметим, что реляционное отношение  $R_1$  позволяет организовать иерархию объектов, а отношение  $R_4$  – иерархию классов одним из ранее описанных способов. Кроме того, атрибут  $c_2$  допускает многократную классификацию объектов по различным признакам, или, иными словами, каждый объект может принадлежать одновременно нескольким классам (свойство полиморфизма), и такая модель более приближена к реальности. Еще одно достоинство заключается в создании условий для оптимизации операций БД на физическом уровне реализации структуры данных (индексирование, поиск, выбор и др.).

Недостатком метода конструирования реляционной модели остается некоторая избыточность в таблицах БД, вызванная различными типами значений параметров, описанных атрибутом  $p_2$ . Вместо одного атрибута  $v_3$  из отношения  $R_3$  потребуется несколько полей для хранения одного и того же значения в различных формах представления. С другой стороны, хранение всех типов данных в универсальной форме (например, текстовой) повлечет за собой необходимость взаимного преобразования данных и повышение вероятности ошибок в операциях чтения/записи.

Созданная методом конструирования модель позволяет извлекать информацию о разнородных объектах БД в рамках одного представления. Множество экземпляров объектов одного класса можно получить путем выборки всех  $o_1$  из  $R_1$  таких, что  $o_2$  соответствует заданному  $c_0$  из  $R_4$ :

$$R_1 [o_1] = \{r: r \in R_1 \wedge (r_1[o_2] \theta r_4[c_0])\} ;$$

Перечень всех параметров, ассоциированных с конкретным объектом, можно получить путем выборки уникальных наборов  $p_1, p_2, p_3$  из  $R_2$  таких, что существуют экземпляры кортежей из  $R_3$ , для которых  $v_2$  принадлежит  $p_0$  из  $R_2$  и  $v_1$  соответствует заданному  $o_0$  из  $R_1$ :

$$R_2 [p_1, p_2, p_3] = \{r: r \in R_2 \wedge (r_3[v_2] \theta r_2[p_0]) \wedge (r_3[v_1] \theta r_1[o_0])\} ;$$

Набор всех значений параметров экземпляра объекта можно получить путем выборки всех  $v_3$  из  $R_3$ , для которых  $v_1$  соответствует заданному  $o_0$  из  $R_1$ :

$$R_3 [v_3] = \{r: r \in R_3 \wedge (r_3[v_1] \theta r_1[o_0])\} .$$

Помимо ограничений целостности реляционных отношений, в работе описаны правила выполнения операций вставки, модификации и удаления экземпляров кортежей для объектов БД и их параметров. Практическая реализация описанных правил и процедур обработки данных довольно сложна и требует тщательного тестирования. Но впоследствии достигается большая экономия времени и труда при добавлении новых объектов, их классов и экземпляров, поскольку проектирование новых объектных структур заменяется на стандартные операции вставки/модификации кортежей в существующей структуре данных посредством вызова имеющихся хранимых процедур. Применение в совокупности методов конструирования, иерархии объектов и их классов и встроенных средств объектно-реляционных СУБД позволяет создавать сложные ИС на уровне предприятия и эффективно управлять ими. Примеры успешной практической реализации ИС приведены в главе 7.

Метод объектного конструирования получил в работе дальнейшее развитие применительно к многомерным наборам данных. В результате разработана *универсальная схема представления хранилища данных стандартными средствами объектно-реляционных СУБД*. Она представляет практическую ценность для небольших предприятий, не имеющих возможность приобрести и установить дорогие OLAP-средства западных производителей.

Построение ИС по принципу хранилища данных, в сравнении с традиционными OLTP-системами, имеет главные отличительные признаки:

- многомерность, временная зависимость наборов данных (OLAP-кубов);
- статичность данных (принцип только накапливания);
- оптимизация запросов (срезов инфокубов) распараллеливанием операций.

В основу реляционного представления хранилища данных положена схема «звезда». Модель включает два типа отношений – отношение фактов  $R_f$  (центр звезды) и отношения  $R_{di}$  по числу измерений (лучи звезды):

$$\begin{aligned} &R_f(a_0, a_1, \dots, a_m, c_1, c_2, \dots, c_n) , \\ &R_{di}(b_{0i}, b_{1i}, \dots, b_{pi}) , \\ &c_1 = b_{01} , \\ &\dots \\ &c_n = b_{0n} , \end{aligned}$$

где  $R_f$  – центральное отношение фактов;  $R_{di}$  – отношение расшифровки  $i$ -го измерения;  $a_1, \dots, a_m$  – базовые атрибуты общих характеристик предметной области;  $a_0$  – ключевой атрибут отношения фактов  $R_f$ ;  $b_{1i}, \dots, b_{pi}$  – атрибуты отношения  $R_{di}$  расшифровки  $i$ -го измерения;  $b_{0i}$  – ключевой атрибут отношения  $R_{di}$  расшифровки  $i$ -го измерения;  $c_1, c_2, \dots, c_n$  – ключевые атрибуты внешних связей с отношениями  $R_{di}$ ;  $c_i = b_{0i}$  – условие соединения отношения фактов  $R_f$  с

$i$ -м отношением  $R_{di}$ ;  $m$  – количество базовых атрибутов предметной области;  $p_i$  – количество атрибутов в отношении  $R_{di}$ ;  $n$  – количество отношений  $R_{di}$  (измерений в многомерной модели);  $i$  – номер измерения,  $1 \leq i \leq n$ .

В структуре отношений  $R_{di}$  может быть организована иерархия измерений одним из способов, описанных выше. Данные в отношениях измерений  $R_{di}$  денормализованы, чтобы ускорить выполнение запросов за счет уменьшения числа таблиц, участвующих в операции соединения. Если все же требуется нормализация отношений  $R_{di}$ , то структура превращается в схему «снежинка».

Схожие объекты объединяются в классы, которые описываются конечным подмножеством атрибутов:

$$C = \{ A_i \mid A_i \in D_A, D_A \in D \},$$

где  $C$  – конечное подмножество классов объектов;  $D_A$  – множество уникальных атрибутов классов объектов.

Все атрибуты объектов и их классов объединены в одно отношение, чтобы было удобно их получать одним запросом. В модель включены только базовые типы данных реляционных СУБД (целый, вещественный, строковый, дата/время). Информационное описание всех объектов всех классов, когда в их состав входят и однозначные, и многозначные атрибуты, имеет вид:

$$\begin{aligned} O &= R' \cup R'' \cup R''' \{ R', R'', R''' \mid \\ R' &\subset \prod_{i=1}^{\infty} D_i', D_i' \in D, D_1' = D_{obj\_id}, \\ R'' &\subset \prod_{i=1}^3 D_i'', D_i'' \in D, D_1'' = D_{obj\_id}, D_2'' = D_{val\_id}, D_3'' = D_{value}, \\ R''' &\subset \prod_{i=1}^4 D_i''', D_i''' \in D, D_1''' = D_{obj\_id}, D_2''' = D_{attr\_id}, D_3''' = D_{val\_id}, D_4''' = D_{value} \}. \end{aligned}$$

где  $O$  – множество объектов всех классов;  $D_{attr\_id}$  – множество однозначных атрибутов;  $D_{obj\_id}$  – множество уникальных идентификаторов объектов класса, в который входит атрибут;  $D_{val\_id}$  – домен, содержащий многозначные атрибуты;  $D_{value}$  – множество значений атрибутов.

Реализация этой реляционной системы хранения объектов приведена в табл. 2. Стержневое отношение  $R_1$  содержит единственную информацию об уникальных идентификаторах объектов и их классов. Объект может существовать сам по себе, даже если ссылки на него отсутствуют в других отношениях. Множество ссылок на объекты одного класса (и всех классов-наследников) на уровне представления можно получить SQL-запросом:

```
select R1.obj_id from R1 where R1.class_id = 'Class1'
```

Таблица 2. Структура объектов с однозначными и многозначными атрибутами

Отношение для хранения уникальных идентификаторов объектов

class_id	obj_id	...
Class1	Object1	
Class1	Object2	
...		
Class2	Object3	
...		
ClassM	ObjectK	

Отношение для хранения однозначных атрибутов (одно для каждого класса)

obj_id	attribute1	...	attributeN
Object1	AttrValue11	...	AttrValue1N
Object2	AttrValue21	...	AttrValue2N
...			
ObjectK	AttrValueK1	...	AttrValueKN

Отношение для хранения единственного многозначного атрибута

obj_id	val_id	value
Object1	1	Value11
Object1	2	Value12
...		
Object2	3	Value23
...		
ObjectK	S	ValueKS

Отношение для хранения нескольких многозначных атрибутов

obj_id	attr_id	val_id	value
Object1	Attribute1	1	Value111
Object1	Attribute1	2	Value112
...			
Object1	Attribute2	3	Value123
...			
Object2	Attribute1	4	Value214
...			
ObjectK	AttributeN	S	ValueKNS

Все однозначные атрибуты класса хранятся в одном отношении  $R_2$ , что увеличивает быстродействие операций поиска, выбора и модификации данных:

`select * from R2 where R2.obj_id = <ссылка на объект>`

С этой же целью создано отношение  $R_3$  для хранения единственного многозначного атрибута. Модификация значения атрибута для одного объекта может быть выполнена SQL-оператором:

`update R3 set R3.value = <значение> where  
R3.obj_id = <ссылка на объект> and R3.val_id = <ссылка на показатель>`

Для нескольких многозначных атрибутов каждому значению соответствует кортеж {obj\_id, attr\_id, val\_id, value} в отношении  $R_4$ , где хранятся все значения атрибутов для всех описанных в системе объектов. Доступ к атрибутам множества объектов возможен при помощи групповой операции SQL, например по изменению данных:

update R4 set R4.value = <значение> where R4.obj\_id = <ссылка на объект> and R4.attr\_id = <ссылка на атрибут> and R4.val\_id = <ссылка на показатель>

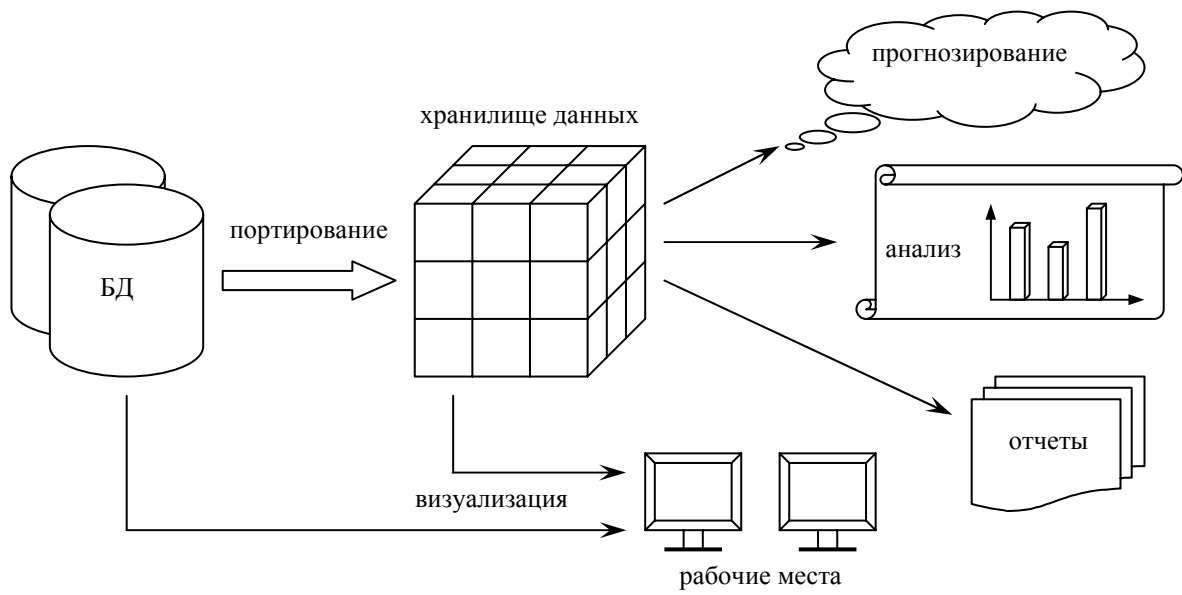
Таким образом при помощи плоских таблиц обеспечивается многомерность наборов данных. Набор показателей любого объекта (срез данных) может быть сохранен в виде отдельного реляционного отношения, что сочетается с принципом только накапливания в хранилищах данных. Кроме того, эта возможность чрезвычайно важна для связи с внешними ИС и интеграции разьединенных детализированных данных – одной из основополагающих концепций хранилищ данных.

Интеграция эксплуатируемых БД и их приложений в единое информационное пространство, а также создание на предприятии системы оперативного получения информации с поддержкой принятия управленческих решений может составлять до 80% общих затрат на разработку ИС в зависимости от принятой *концепции интеграции*:

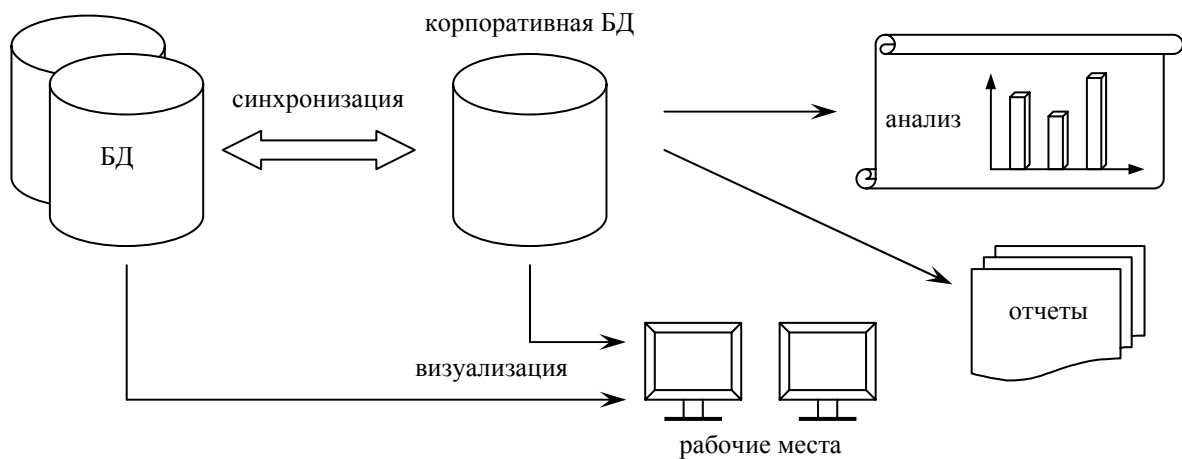
1. Объединение всех ресурсов предприятия в единое хранилище данных по OLAP-технологии и создание системы поддержки принятия решений (рис. 6,а). Это прогрессивное направление предполагает кардинальную перестройку традиционно сложившихся информационных процессов, большие затраты и долгосрочную перспективу (2-3 года), но после внедрения дает наибольший эффект. Кроме того, всегда присутствует риск того, что интеграционное решение не покроет на 100% все области автоматизации и не окупит затраты на внедрение. Поэтому такой сценарий развития может позволить себе только крупное предприятие с хорошей рентабельностью.

2. Концепция корпоративной БД (рис. 6,б). Построение корпоративной БД объектно-реляционного типа, дополнение существующих приложений модулями доступа к общим информационным ресурсам, разработка новых интеграционных приложений для визуализации, агрегирования, анализа корпоративных данных, формирования отчетности. Этот умеренный сценарий развития ИС характеризуется постепенным вложением средств и доступен любому предприятию, но рассчитан на длительную реализацию.

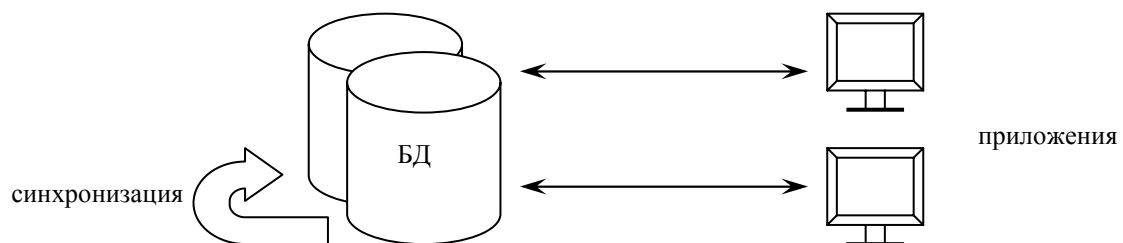
3. Адаптация эксплуатируемых БД и внешних приложений, обеспечение синхронизации данных в различных структурах БД (рис. 6, в). Для этого потребуется создание механизмов обмена с внешними источниками данных, модификация хранимых процедур и гетерогенных запросов доступа к разнородным данным для их синхронизации. Такой сценарий обеспечивает минимум затрат на интеграцию ИС, поскольку отлаженные программные решения остаются в эксплуатации и наращиваются новыми модулями. Главным недостатком является затрудненность получения неоднородной информации, ее статистического анализа, формирования общих отчетов в гетерогенной среде, где применяется продукция разных поставщиков.



а) концепция единого хранилища данных



б) концепция корпоративной БД



в) адаптация существующих БД и их приложений

Рис. 6. Концепции интеграции ИС

**В четвертой главе** представлены *эффективные методы и алгоритмы обработки информации в клиентских приложениях*: процедуры портирования данных в новые структуры, методы оптимизации SQL-запросов к серверу БД, нетрадиционные способы генерации отчетов, частные задачи хранения нестандартных типов данных, механизмы обращения к БД через универсальный web-интерфейс «тонких» клиентов, представление графической информации в БД и автоматизированное проектирование.

Стратегии реорганизации и перевода ИС на другую архитектуру могут быть разными по масштабам – революционные, означающие необратимый процесс реструктуризации, и эволюционные, т. е. поэтапные и менее «болезненные». В отличие от революционной, эволюционная модернизация ИС может быть растянута по времени и затратам, если ее удастся разбить на логические этапы.

*Портирование накопленных данных в новые структуры БД* – очень ответственная задача. Унаследованная информация представляет собой самое ценное из устаревшей ИС, но процедуру переноса данных редко удается выполнить безболезненно по разным причинам – неудачно спроектированная структура данных, нарушение ссылочной целостности, несовместимость методов хранения и обработки данных в старой и новой БД. Решение проблем миграции данных представлено разными методами и процедурами:

- Перенос данных методом обратного проектирования – наиболее быстрый и распространенный способ переноса данных «как есть» с заменой типов данных на эквивалентные согласно приведенным правилам.
- Замена логических типов в приложении и на уровне хранимых процедур.
- Устранение несовместимости русских кодировок в строковых и мемо-полях.
- Автоматическое конвертирование форматированных текстов.
- Исправление ошибок в датах, в т. ч. «проблемы 2000 года».

Использование предложенных процедур анализа и обработки данных позволяет скорректировать до 95-98% ошибок при переносе данных из одной структуры в другую, уменьшая тем самым потерю информации.

Для обеспечения совместной работы приложений в гетерогенной среде разработаны процедуры синхронизации данных между разными структурами внутри одной БД, разными экземплярами БД (рабочая и архивная копии, локальные БД на рабочих местах), а также между разными типами СУБД:

- соединение таблиц данных стандартными средствами SQL;
- обновление данных последовательным сравнением ключевых полей;
- обновление связки таблица–справочник при изменяемых внешних ссылках;
- синхронизация больших наборов данных, когда не действуют стандартные методы поиска.

В клиент-серверной архитектуре эффективность работы приложений БД во многом зависит от производительности выполнения запросов к серверу. Многие вопросы эффективности запросов могут быть решены на стадии



проектирования структуры БД. Для ряда частных задач, встречающихся на практике, предложены *нестандартные методы хранения данных*:

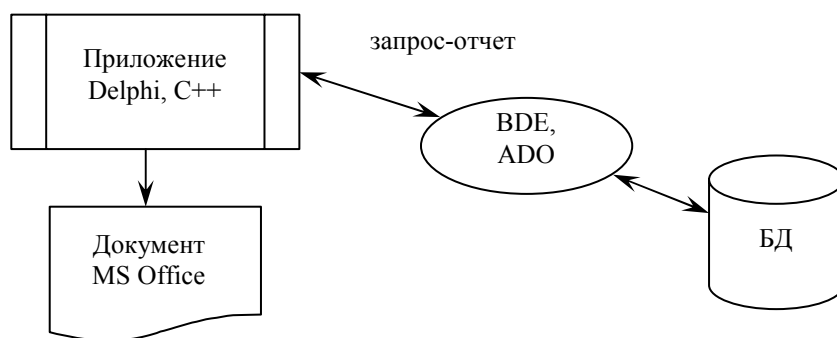
- универсальный метод описания диапазонов данных;
- объединение большого количества однопольных атрибутов в целях уменьшения ширины таблицы;
- разделение множества данных на подмножества для эффективного поиска в любых позициях строковых полей;
- репликация подмножества данных – дублирование части информации в отдельном поле для поиска и обработки длинных текстов и CLOB-объектов;
- частичные отношения – структура, позволяющая эффективно реализовывать реляционные операции соединения таблиц;
- хранение в БД результатов выполнения относительно статичных запросов;
- хранение больших двоичных BLOB-объектов в сжатом виде;
- денормализация реляционных отношений, чтобы исключить соединение со справочными таблицами и отслеживать динамику их изменения.

По результатам выполнения пользовательских запросов генерируются отчеты, которые могут быть «защиты» в приложение, экспортироваться в шаблон документа (MS Office, HTML, XML) или создаваться динамически средствами VBA. В работе дана краткая характеристика современных средств генерации отчетов, в том числе через универсальный web-интерфейс «тонких» клиентов, но главное внимание в этой чрезвычайно актуальной области уделено *нетрадиционным технологиям формирования отчетных документов* на основе объектного подхода к управлению БД. На рис. 7 показаны возможные механизмы получения выборки данных для отчета с использованием технологии OLE DB.

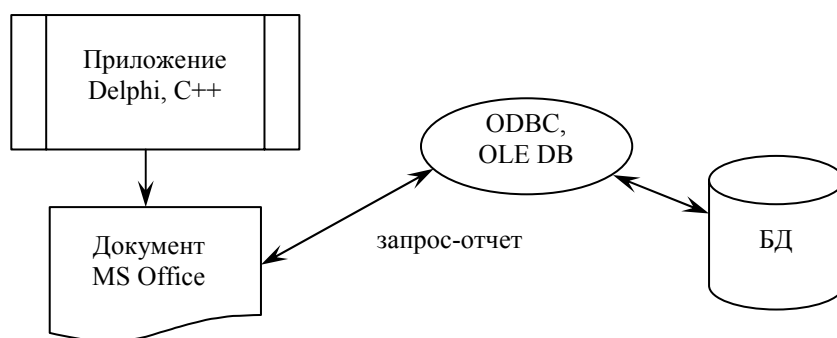
В первом классическом варианте (рис. 7,а) для получения выборки данных используется тот же драйвер (BDE, ADO), с которым работает приложение. При этом связь с отчетным документом односторонняя, и хотя пользователь имеет возможность вносить некоторые изменения в полученный отчет, результаты его труда не могут быть сохранены в БД.

Для преодоления этих недостатков введена новая управляющая связь объектов документа с объектами БД. Согласно схемы (рис. 7,б), запрос в формате MS Query хранится в шаблоне документа MS Office, а при его открытии (активации) происходит обращение к внешнему источнику данных через драйвер ODBC и обновление соответствующих полей данных.

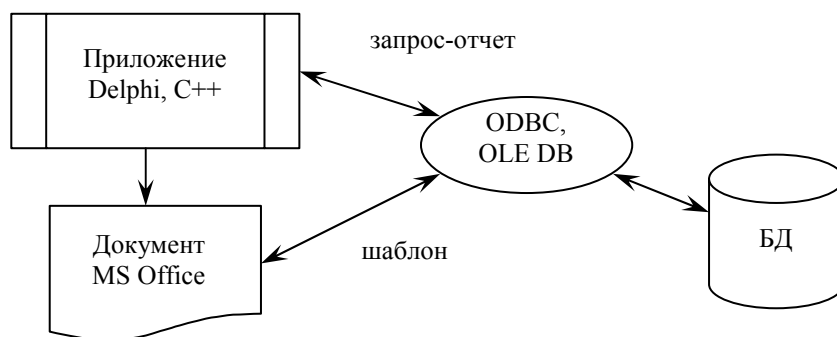
Если шаблон хранить не в локальном файле, а как OLE-объект, внедренный в BLOB-поле, то для его активации требуется дополнительное обращение к БД (рис. 7, в), и после обработки на стороне клиента результат также будет сохранен обратно в БД как OLE-объект. Любая схема реализации требует совместимости всех программных средств, участвующих в цепочке «приложение–запрос–документ».



а)



б)



в)

Рис. 7. Схемы получения выборки данных для отчета

Практика сопровождения ИС на предприятиях показывает, что невозможно предусмотреть формы отчетов на все случаи жизни. В процессе эксплуатации меняются требования, нормативы, кадровый состав и т. п. Поэтому иметь возможность получения произвольного отчета – объективное требование современных ИС. Для этого необходим гибкий механизм построения запросов. В общем виде SQL-запрос имеет следующую структуру:

```

select <список полей>
from <список таблиц>
where <условия соединения таблиц> and <условия фильтрации записей>
group by <перечень групп>
order by <порядок сортировки>

```

В такой неопределенной форме запрос нельзя интегрировать в приложение. Другая проблема в том, что пользователь, формулирующий вопрос, является специалистом в предметной области и не имеет представления о SQL. Поэтому предлагается подход, который позволяет специалисту гибко формулировать запросы к БД, используя понятия предметной области, и не требует знания SQL и структуры таблиц. Список полей, таблиц и условия их соединения можно определить на стадии разработки приложения, а на этапе выполнения предоставить пользователю выбор условий фильтрации одним из двух способов – принадлежность множеству или вхождение в интервал. В зависимости от типа поля (числовой, строковый, дата, внешняя ссылка) реализуется одна из форм SQL-операторов.

В табл. 3 показан практический пример задания параметров для условий фильтрации в SQL-запросе, в результате которого выбирается подвесное оборудование в цехах 1, 3 и 7, у которого в период 2001 года проводились ремонтные мероприятия по замене болтовых или других крепежных деталей из латуни. Полученная выборка данных экспортируется в универсальный шаблон Excel, в котором предусмотрены столбцы для всех атрибутов, и чтобы скрыть лишние колонки с пустыми значениями, выполняется их постобработка макросом VBA в цикле:

```

For i = 1 To Columns.Count
    If ActiveCell.Value = "" Then
        ActiveCell.EntireColumn.Hidden = True
    End If
    Application.Goto Reference:="R[0]C[1]"
Next i

```

Таблица 3. Задание условий фильтрации для произвольного запроса

Атрибуты	Параметры выбора	Условия фильтрации в SQL-запросе
Цех	1, 3, 7	and cex in (1, 3, 7)
Тип оборудования	подвесное	and lower(model) = "подвесное"
Дата ремонта	1.01.2001:31.12.2001	and date_fact >= mdy(1,1,2001) and date_fact <= mdy(12,31,2001)
Проведенные мероприятия	крепеж, болт	and (lower(action) matches "*крепеж*" or lower(action) matches "*болт*")
Материал внутренних устройств	латун	and id_mat in (select id from spr_mat where lower(mat) matches "*латун*")

Средства VBA используются также при организации графической БД для доступа к внутренним объектам AutoCAD и управления ими из внешнего приложения. Совместная работа внешнего приложения и среды AutoCAD возможна по одной из двух технологий:

- 1) проект VBA внедрен в чертеж AutoCAD и загружается вместе с файлом;
- 2) проект VBA сохранен в отдельном файле (DVB), и с его помощью можно управлять загрузкой сессии AutoCAD и любых чертежей.

Связь внешнего приложения, VBA-проекта и сессии AutoCAD, а также доступ к внутренним объектам чертежа поддерживаются при помощи механизма манипулирования объектами ActiveX Objects (рис. 8). Наличие такой технологии не отменяет, а наоборот, дополняет действующие механизмы обмена данными между БД и внешними приложениями (BDE, DAO, ODBC). *Управление графическими построениями* реализуется двумя способами:

- 1) загрузка готового чертежа и его модификация средствами VBA;
- 2) вычерчивание с чистого листа по числовым параметрам, хранящимся в БД.

*Параметризация служит эффективным способом описания чертежа*, т. к. графический объект модифицируется путем изменения набора атрибутов, а вычерчивается автоматически. Технология параметризации показана на примере построения отрезка прямой линии по двум точкам функциями VBA:

```
Function DrawLine(StartPoint, EndPoint) As Object
    Dim LineObject As AcadLine
    Set LineObject = ThisDrawing.ModelSpace.AddLine(StartPoint, EndPoint)
    ThisDrawing.Application.Update
    ThisDrawing.Application.ZoomAll
End Function
```

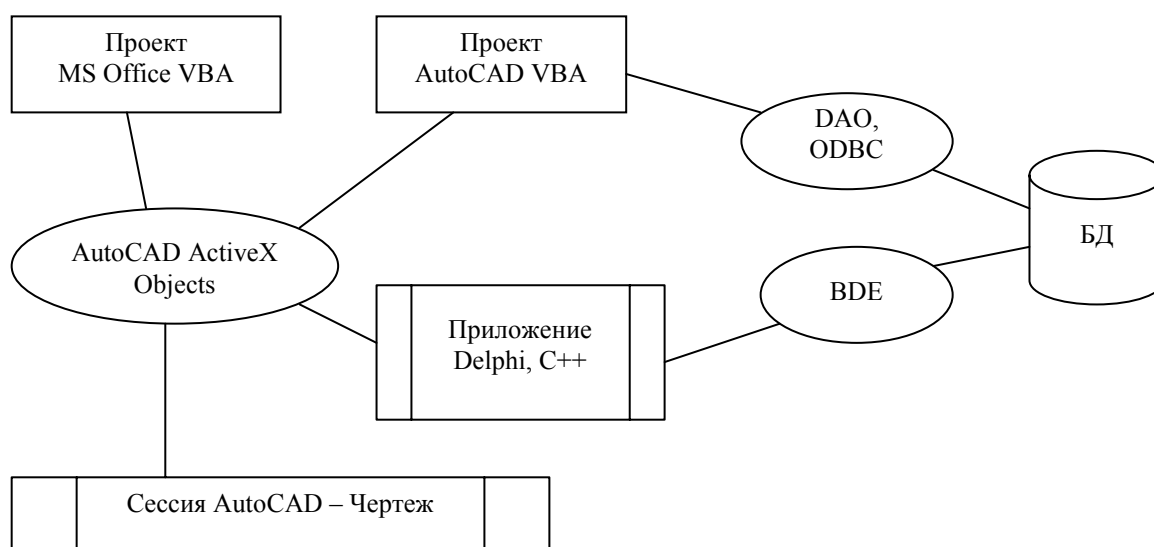


Рис. 8. Доступ к внутренним объектам AutoCAD из внешних приложений

Аналогичным образом определяются функции VBA для параметрического построения других графических примитивов, блоков и даже графического объекта в целом. Построение чертежа сводится к последовательному вызову соответствующих функций с подстановкой фактических параметров:

```
Dim L As AcadLine
Dim StartPoint(0 To 2) As Double
Dim EndPoint(0 To 2) As Double
StartPoint(0) = 1: StartPoint(1) = 1: StartPoint(2) = 0 ' координаты начальной точки
EndPoint(0) = 100: EndPoint(1) = 100: EndPoint(2) = 0 ' координаты конечной точки
Set L = DrawLine(StartPoint, EndPoint) ' вызов функции построения линии
```

В главе 7 представлена практическая реализация параметрической модели управления графическими объектами в среде AutoCAD на примере листовой конструкции сосудов, резервуаров и емкостей цилиндрической формы.

**В пятой главе** сформулированы принципы проверки качества ПО, дана классификация методов тестирования ИС и приложений БД – интегральных, компонентных, корреляционных, эталонных, получены *результаты экспериментальной проверки методов проектирования БД*, описанных в главе 3:

- 1) сравнительное тестирование методов организации иерархии данных;
- 2) измерение производительности БД с различной архитектурой.

Для проверки относительных преимуществ классических и нетрадиционных подходов к организации иерархии данных поставлены 2 категории вычислительных экспериментов по оценке характеристик производительности СУБД:

- 1) измерение производительности операций доступа к данным (scan) в зависимости от глубины вложенности и размера иерархических таблиц (рис. 9);
- 2) измерение операций соединения таблиц (join) при получении потомков на различных уровнях вложенности.

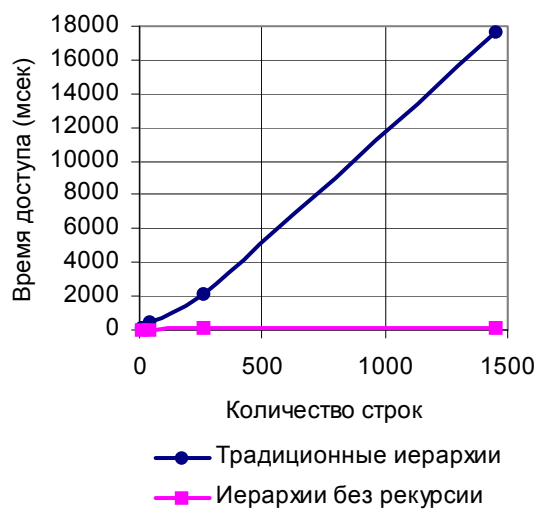
В целях удобства проведения сравнительных тестов и анализа результатов все варианты иерархической структуры сгруппированы по критерию наличия рекурсии в реляционных отношениях и SQL-запросах:

- 1) традиционные иерархические структуры с рекурсивными связями;
- 2) варианты иерархии данных без рекурсивности.

Относительные показатели производительности (рис. 9, в, г) вычислялись для оценки различных вариантов иерархии в терминах удельных единиц измерения (сек/уровень, сек/запись):

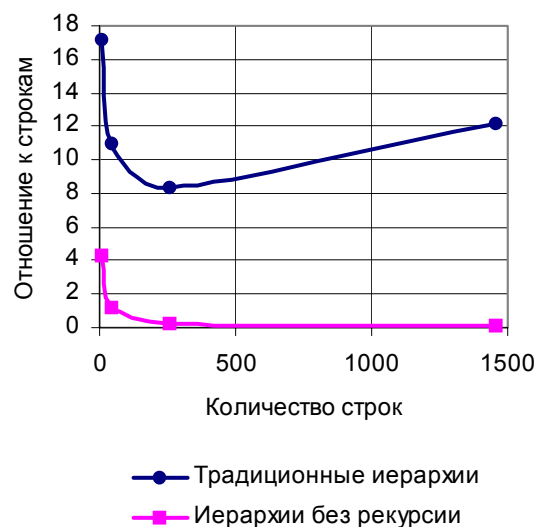
$$R_{y1} = T_{II} / N_y, \quad R_{z1} = T_{II} / N_z,$$

где  $R_{y1}$  – отношение времени к числу уровней вложенности по результатам одного теста;  $R_{z1}$  – отношение времени к количеству записей по результатам одного теста;  $T_{II}$  – измеренное время выполнения операции (сек);  $N_y$  – глубина вложенности (число уровней);  $N_z$  – размер таблицы (число записей).



а) зависимость времени доступа от глубины иерархии

б) зависимость времени доступа от размера таблицы данных



в) отношение времени доступа к числу уровней иерархии

г) отношение времени доступа к количеству строк таблицы данных

Рис. 9. Сравнение производительности операции доступа к иерархическим данным (scan)

Все виды тестов проводились на стабилизированной системе по критерию разброса результатов не более 2%. Значения для каждого теста получены путем статистического усреднения результатов нечетного числа экспериментов (не менее 3 с учетом того, что крайние максимальное и минимальное значения из выборки исключаются). Результаты тестирования показали, что отсутствие рекурсии дает улучшение производительности от 4 до 125 раз в зависимости от глубины иерархии и размера таблицы данных.

Вторая серия вычислительных экспериментов посвящена тестированию ИС как готовых программно-аппаратных решений при различной конфигурации сети – двухуровневой и с промежуточным слоем с целью оценки их производительности и работоспособности, а также выявления узких мест:

- 1) измерение времени выполнения операции доступа к данным (select) в зависимости от числа клиентских запросов;
- 2) измерение времени выполнения операции модификации данных (update) с использованием транзакций в зависимости от числа клиентов.

В трехуровневой конфигурации с ростом нагрузки (одновременно работающих сессий) падение производительности (увеличение времени отклика) незначительно, что говорит о хорошей масштабируемости и устойчивости работы в многопользовательской среде. Анализ полученных результатов выявил возможность оптимизации системы за счет перераспределения нагрузки между сервером и клиентом. Несмотря на то, что больше половины нагрузки приходится на сервер, общая загруженность его ресурсов не превышает 65%.

Измерение показателей загруженности и анализ производительности позволяют прогнозировать поведение системы в будущем и заранее планировать мероприятия по модернизации оборудования и ПО.

**Шестая глава** посвящена вопросам *информационной безопасности и администрирования БД*. Источники угрозы и уязвимости систематизированы по степени ущерба и механизмам проникновения. Дана классификация технических средств и организационных мероприятий по защите информации от несанкционированного доступа (рис. 10). Сформулированы правила информационной безопасности и антивирусной защиты применительно к любому предприятию. Перечислены требования и рекомендации к формированию паролей и обращению с ними. Отмечены источники угрозы НСД и приведены методы управления доступом, специфичные для баз данных.

Ни один из перечисленных способов в отдельности не может гарантировать 100% защиты информации. Только их комплексным применением можно достичь наилучшего уровня защищенности. Но совершенствование средств защиты информации стимулирует появление новых методов НСД. Нельзя учесть все варианты возможного вторжения, поэтому в системе безопасности нужно предусматривать экстренное реагирование на внештатные ситуации. Пример такого программно-аппаратного решения представлен в главе 7.

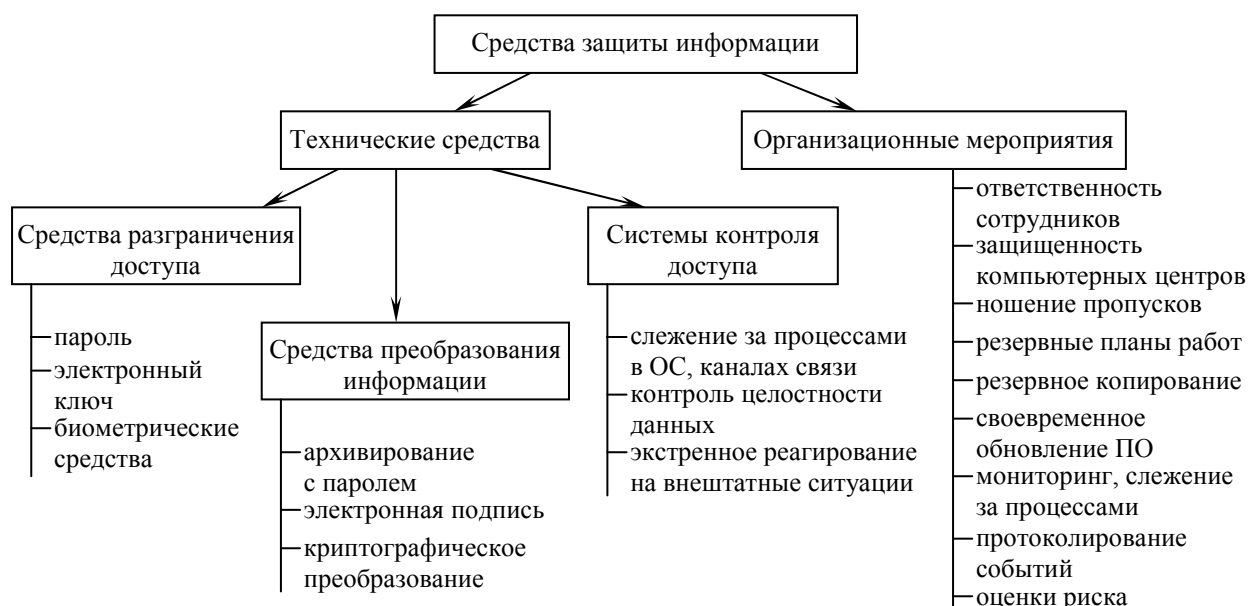


Рис. 10. Классификация средств защиты информации

**В седьмой главе** представлены результаты практической реализации методов проектирования ИС и приложений БД на предприятиях северо-западного региона России.

*Автоматизированная ИС учета состояния технологического оборудования* разработана в СУБД Informix и функционирует в сети крупнейшего в регионе завода «ПО КИНЕФ» (г. Кириши). БД включает: электронные паспорта оборудования с чертежами, справочную подсистему, результаты замеров степени коррозии и архив всех ремонтов с 1986 г., оперативное планирование мероприятий по технадзору, инженерные расчеты и прогнозирование ресурса остаточной работоспособности оборудования, отчетную документацию.

Классификация оборудования имеет иерархическую структуру. Каждый класс определяется своим набором параметров согласно концепции объектно-реляционного конструирования. Информационное наполнение электронных паспортов проводилось специально разработанными процедурами миграции данных из источников, накопленных в предыдущей версии файл-серверной БД.

Чертежи оборудования, выполненные в AutoCAD, хранятся в BLOB-полях электронных паспортов в векторных форматах DWG и WMF. Управление средой AutoCAD и графическими построениями производится из клиентских приложений средствами VBA. Конструкция отдельных видов оборудования (сосудов, резервуаров, емкостей, котлов, теплообменников) состоит из типовых элементов, сваренных из листового материала, – цилиндрических обечаек, эллиптических днищ, штуцеров. Графическая модель этих объектов представлена параметрически в виде развертки.



Автоматизация учета технического состояния дает возможность увеличить периодичность освидетельствований и ремонтов для оборудования, средний возраст которого превышает 20 лет (рис. 11), но которое имеет относительно небольшую скорость коррозии по результатам замеров, что напрямую ведет к снижению материальных и финансовых затрат на эксплуатацию.

*Интегрированная ИС комплексного обследования ГПМ* разработана и функционирует на инженерно-экспертном предприятии по экспертизе промышленной безопасности подъемных сооружений ЗАО «РАТТЕ». В ее состав входят приложения: АРМ обследования грузоподъемных кранов, комплексное обследование крановых путей, автоматизация документооборота.

База данных комплексных обследований реализована в СУБД MS FoxPro и содержит паспортные и справочные данные грузоподъемных кранов, характеристики крановых путей, сведения об экспертной комиссии, дефектную ведомость, нивелировки крановых путей, акт комплексного обследования. Результаты замеров сводятся в электронные таблицы Excel по технологии шаблона, где строятся графики нивелировок. Ведомость дефектов формируется как совокупность дефектных узлов и мероприятий по их устранению. Акт комплексного обследования с дефектной ведомостью на ремонт экспортируется в MS Word по технологии шаблона, а после окончательной корректировки инженером-экспертом сохраняется в БД как внедренный OLE-объект.

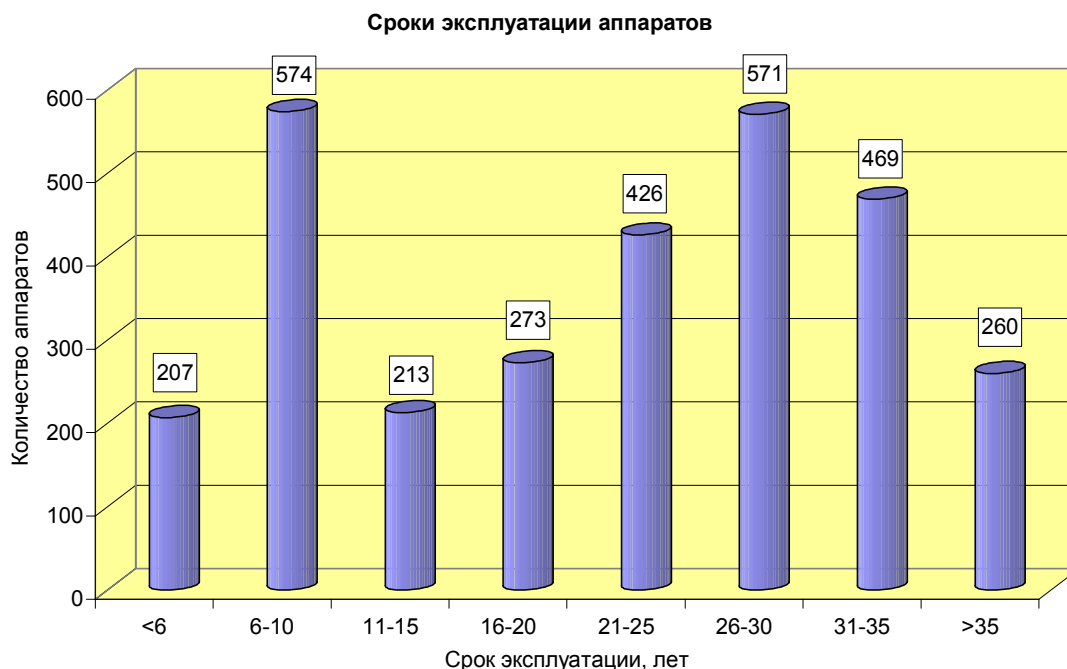


Рис. 11. Сведения о сроках эксплуатации и количестве эксплуатируемого оборудования

Внедрение разработанных программных решений в ЗАО «РАТТЕ» позволило автоматизировать учет, снизить трудоемкость и улучшить качество подготовки экспертных заключений и прочей технической документации. Кроме того, накапливаемая в БД совокупная информация по всем объектам учета о дефектах и мероприятиях по их устранению облегчает контроль за исполнением регламентных работ и позволяет проводить анализ статистики наиболее частых отказов узлов оборудования, эксплуатируемого на разных предприятиях.

*Программно-аппаратный комплекс информационной защиты от НСД и экстренного реагирования на внештатные ситуации является эффективным средством поддержки конфиденциальности информации. Он спроектирован в среде Visual C++ как специализированный продукт и адаптирован к условиям компьютерной сети ЗАО «АМН». Минимальный комплекс необходимых аппаратных средств включает следующее оборудование: информационный сервер с установленным ПО системы безопасности и источником бесперебойного питания для страховки от потери питания в момент НСД, датчики слежения в серверном помещении с подключением к одному из портов сервера, устройство ввода на сменном носителе с подключением к одной из рабочих станций для ввода ключа идентификации доступа к серверу.*

Начальный запуск ПО системы безопасности производится с сервера. В дальнейшем активация/деактивация процесса слежения возможна двумя способами – с сервера или удаленно с рабочей станции, к которой подключено устройство ввода ключа идентификации. После запуска система безопасности начинает процесс слежения за сигналами с датчиков проникновения с протоколированием всех событий. В случае внешнего НСД к серверу и при отсутствии ключа идентификации в устройстве ввода на рабочей станции активируется механизм экстренной защиты конфиденциальной информации.

Применение разработанных алгоритмов и программ позволило обеспечить высокий уровень информационной безопасности в условиях совместной работы с документами общего пользования. Подобные нестандартные решения всегда уменьшают риск обойти такую систему защиты.

**В заключении** подчеркивается прогрессивность предложенных методов и технологий проектирования баз данных в новых экономических условиях, когда создание информационных систем поставлено на индустриальную основу. Сформулированы актуальные направления дальнейших научных исследований в области разработки ИС и приложений БД.

**В приложениях** приведены исходные тексты процедур обработки данных, структура прототипа промышленной БД учета состояния технологического оборудования, а также документы, подтверждающие практическую реализацию результатов научных исследований на производственных предприятиях и в учебном процессе.

## ВЫВОДЫ

Основные научные положения диссертационной работы и их практическая реализация представляют собой дальнейшее развитие теории проектирования баз данных и решение научной проблемы – разработки эффективных методов построения информационных систем и приложений баз данных, что имеет практическую ценность для задач управления сервисными производственными системами.

В диссертации получены следующие новые теоретические и практические результаты:

1. В результате проведенных исследований современных технологий проектирования баз данных разработаны новые схемы описания производственных объектов для организации на предприятии единого информационного пространства и более эффективного решения задач технического обслуживания производства.

2. На основе обобщения существующих моделей разработана информационная модель объектов грузоподъемного оборудования, которая служит прототипом при проектировании более сложных иерархических моделей других производственных объектов.

3. Решена задача автоматизации многофункционального учета производственных объектов, что позволило неограниченно увеличить количество разнородных параметров, характеризующих техническое состояние объектов, и тем самым повысить эффективность обслуживания производства.

4. Разработаны средства многоуровневой классификации производственных объектов. Предложены новые методы организации иерархии, не содержащие рекурсию. Результаты проведенного вычислительного эксперимента показали улучшение производительности безрекурсивных методов от 4 до 125 раз в зависимости от глубины иерархии и объема данных.

5. Предложена методика конструирования информационно-управляющих моделей производственных объектов любой сложности, которая позволила решить задачу единого информационного учета эксплуатируемого на предприятии основного и вспомогательного технологического оборудования.

6. На основе разработанных информационных моделей построена система управления мероприятиями по экспертизе промышленной безопасности и ремонту производственных объектов, которая успешно реализована на практике и эксплуатируется на предприятиях северо-западного региона России.

7. Разработаны эффективные алгоритмы преобразования данных, что позволило практически исключить потери накопленной информации при ее миграции в новые управляющие структуры. Реализованы новые средства хранения и обработки нестандартных типов данных, встречающихся в задачах

управления информационными потоками, которые невозможно реализовать традиционными методами.

8. Исследованы потребности современных предприятий в автоматизации документооборота. Разработаны новые методы формирования отчетной документации по результатам произвольных запросов к базе данных.

9. Решена задача параметризации технических чертежей производственных объектов, хранимых в базе данных, для более эффективного управления графическими построениями.

10. В результате анализа источников угрозы информационной безопасности и существующих технических средств защиты в базах данных разработана и внедрена система управления с более высоким уровнем защиты информации от несанкционированного доступа.

Хранилища и базы данных служат основой современных информационных систем. Накапливаемая в них информация представляет собой наиболее ценный материал, и эффективные методы ее хранения и обработки чрезвычайно важны для получения новых знаний.

#### ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Мещеряков С.В., Розанов Л.Н., Печатников Ю.М. Организация базы данных элементов и материалов вакуумных систем / Вакуумная наука и техника: Тез. докл. науч.-техн. конф. с участием зарубежных специалистов.– Гурзуф, 1994.– 0,06 п. л.
2. Мещеряков С.В., Печатников Ю.М. Методика преподавания интеллектуальных САПР в машиностроении / Совершенствование учебного процесса при подготовке инженеров-механиков: Сб. науч.-методич. тр.– СПб: СПбГТУ, 1994.– 0,13 п. л.
3. Гончаров Б.Ф., Горинштейн А.М., Мещеряков С.В., Печатников Ю.М., Петров И.П., Розанов Л.Н. Машиностроительная информатика. Вычислительная техника: Лаб. практикум.– СПб: СПбГТУ, 1994.– 6,75 п. л.
4. Донская М.М., Мещеряков С.В., Печатников Ю.М., Розанов Л.Н., Сказываев В.Е. Программное обеспечение курсового проектирования: Аннотированный каталог учебных программных средств, Вып. 3.– СПб: СПбГТУ, 1995.– 0,06 п. л.
5. Мещеряков С.В., Печатников Ю.М. Информационная система вакуумного оборудования и материалов / Вакуумная техника и технология.– 1995.– № 1/2.– 0,31 п. л.
6. Донская М.М., Мещеряков С.В., Печатников Ю.М. Учебная САПР вакуумных систем / Вакуумная техника и технология.– 1995.– № 3.– 0,25 п. л.

7. Донская М.М., Мещеряков С.В. Архитектура автоматизированного рабочего места конструктора: Метод. указ. к лаб. раб. по курсу «Информационная технология».– СПб: СПбГТУ, 1995.– 0,5 п. л.
8. Мещеряков С.В. Автоматизация компоновки в машиностроении / Инновационные наукоемкие технологии для России: Тез. докл. Российск. науч.-техн. конф., Ч. 4.– СПб: СПбГТУ, 1995.– 0,06 п. л.
9. Дзельтен Г.П., Мещеряков С.В., Печатников Ю.М., Розанов Л.Н., Розанов С.Л., Солодилова Н.А. Разработка методов автоматизированного проектирования и анализа вакуумных систем: Отчет о НИР, № ГР 0196.0002132.– СПб: СПбГТУ, 1995.– 2,5 п. л.
10. Мещеряков С.В., Розанов Л.Н., Розанов С.Л., Сказываев В.Е. Разработка информационных средств по вакуумной технике и технологии: Отчет о НИР, № ГР 0196.0005212.– СПб: СПбГТУ, 1995.– 1,88 п. л.
11. Гончаров Б.Ф., Донская М.М., Мещеряков С.В., Розанов Л.Н., Трубин И.А., Харисов С.А. Информатика. Технические средства САПР: Лаб. практикум.– СПб: СПбГТУ, 1996.– 4,5 п. л.
12. Солодилова Н.А., Мещеряков С.В. База данных элементов вакуумного оборудования / Прогрессивные конструкции и технологии в машиностроении: Сб. науч. работ студентов и аспирантов, № 7.– СПб: СПбГТУ, 1996.– 0,19 п. л.
13. Мещеряков С.В. Графическая система имитационного моделирования транспортных потоков автоматизированного производства / Фундаментальные исследования в технических университетах: Материалы I Всероссийской конф.– СПб: СПбГТУ, 1997.– 0,06 п. л.
14. Петров А.Ю., Мещеряков С.В. Организация электронной почты в локальной компьютерной сети / Современные научные школы: перспективы развития: Тез. докл. науч. конф.– СПб: СПбГТУ, 1998.– 0,13 п. л.
15. Иванов С.А., Мещеряков С.В. Архитектура компьютерной сети Интранет для поддержки баз данных / XXVII Неделя науки СПбГТУ: Материалы науч.-техн. конф.– СПб: СПбГТУ, 1998.– 0,06 п. л.
16. Рябцев М.В., Мещеряков С.В. Организация доступа к базам данных через Интернет / XXVII Неделя науки СПбГТУ: Материалы науч.-техн. конф.– СПб: СПбГТУ, 1998.– 0,06 п. л.
17. Мещеряков С.В. Изучение электронной почты в локальной компьютерной сети: Метод. указ. к лаб. раб. по курсу «Информационные технологии».– СПб: СПбГТУ, 1998.– 1,13 п. л.
18. Мещеряков С.В. Методические указания по выполнению магистерских диссертаций: Учеб. пособие.– СПб: СПбГТУ, 1999.– 0,75 п. л.
19. Гусаров А.Л., Мещеряков С.В. Автоматизация управления единой информационной службой предприятия в Интернет / XXIX Неделя науки СПбГТУ: Материалы науч.-техн. конф.– СПб: СПбГТУ, 2000.– 0,06 п. л.

20. [Андреев Л.Н.], Бортяков Д.Е., Мещеряков С.В. Основы автоматизированного проектирования: Конспект лекций.– СПб: СПбГТУ, 2001.– 4,88 п. л.
21. [Андреев Л.Н.], Бортяков Д.Е., Мещеряков С.В. Системы автоматизированного проектирования: Учеб. пособие.– СПб: СПбГПУ, 2002.– 4,88 п. л.
22. Мещеряков С.В. О новых технологиях разработки приложений баз данных / Высокие интеллектуальные технологии образования и науки: Материалы IX Междунар. науч.-метод. конф.– СПб: СПбГПУ, 2002.– 0,06 п. л.
23. Мещеряков С.В. Проблемы и средства управления приложениями баз данных / Фундаментальные исследования в технических университетах: Материалы VI Всероссийской конф.– СПб: СПбГПУ, 2002.– 0,06 п. л.
24. Холодовский О.В., Мещеряков С.В., Иванов В.М. Методы снижения загруженности компьютерной сети / Фундаментальные исследования в технических университетах: Материалы VI Всероссийской конф.– СПб: СПбГПУ, 2002.– 0,06 п. л.
25. S.V. Mescheryakov. A Successful Implementation of a Data Structure for Storing Multilevel Objects with Varying Attributes. IBM, Informix Developer Zone, 2002.– 0,94 п. л.  
<http://www7b.software.ibm.com/dmdd/zones/informix/library/techarticle/0212mescheryakov/0212mescheryakov.html>
26. Мещеряков С.В. Повышение эффективности ремонта оборудования / Оптимизация транспортных машин: Сб. науч. тр.– Ульяновск: УЛГТУ, 2003.– 0,38 п. л.
27. Мещеряков С.В. Технические средства защиты в базах данных / Фундаментальные исследования в технических университетах: Материалы VII Всероссийской конф.– СПб: СПбГПУ, 2003.– 0,06 п. л.
28. Мещеряков С.В., Иванов В.М., Чалей И.В. Автоматизированный учет состояния технологического оборудования / Научно-технические ведомости СПбГПУ.– 2003.– № 2.– 0,56 п. л.
29. Мещеряков С.В. Прогнозирование остаточного ресурса технологического оборудования / Научно-технические проблемы прогнозирования надежности и долговечности конструкций и методы их решения: Доклад V Международной конф.– СПб: СПбГПУ, 2003.– 0,13 п. л.
30. Мещеряков С.В. Проблемы организации графических баз данных / Современные проблемы информатизации в системах моделирования, программирования и телекоммуникациях: Сб. трудов IX Международной открытой науч. конф.– Воронеж: ВорГТУ, 2004.– 0,06 п. л.  
<http://www.vsi.ru/~sbph/spi/archives/SPI20043.RAR>
31. Мещеряков С.В. Современные требования компьютерной подготовки выпускников технических вузов / Формирование профессиональной культуры специалистов XXI века в техническом университете: Тез. докл. 4-й

- Международной науч.-практич. конф.– СПб: Изд-во СПбГПУ, 2004.– 0,13 п. л.
32. Мещеряков С.В. Методы представления графической информации в приложениях баз данных / Информационные технологии моделирования и управления: Международный сб. науч. трудов, Вып. 15.– Воронеж: Изд-во «Научная книга», 2004.– 0,44 п. л.  
<http://www.vsi.ru/~sbph/itmu/archives/sysi0415.rar>
33. S.V. Mescheryakov. Performance Comparison of Various Hierarchical Structures in Database Systems. Proceedings of the 30th International Conference for the Resource Management and Performance Evaluation of Enterprise Computing Systems. Las Vegas, USA, 2004.– 0,63 п. л.  
<http://www.cmg.org/membersonly/2004/papers/4217.pdf>
34. Мещеряков С.В., Иванов В.М. Эффективные технологии создания информационных систем.– СПб: Издательство «Политехника», 2005.– 19,5 п. л.
35. Мещеряков С.В., Иванов В.М. Построение объектно-реляционных моделей баз данных с произвольным набором атрибутов / Системы управления и информационные технологии.– 2005.– № 4 (21).– 0,75 п. л.  
<http://www.vsi.ru/~sbph/suit/an200504.htm>
36. Мещеряков С.В. Информационная система документооборота.– Каталог программных систем, Арбинада, 2006.– 0,06 п. л.  
<http://www.arbinada.com>
37. Мещеряков С.В. Успешная реализация структуры данных для хранения многоуровневых объектов с произвольным набором атрибутов: Перев. с англ.– Архитекторы информационных систем, Арбинада, 2006.– 0,94 п. л.  
<http://www.arbinada.com/modules.php?name=Content&pa=showpage&pid=88>

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

АРМ – автоматизированное рабочее место  
АСТПП – автоматизированная система технической подготовки производства  
БД – база данных  
ГПМ – грузоподъемный механизм  
ГПО – грузоподъемное оборудование  
ИС – информационная система  
ИТ – информационные технологии  
НСД – несанкционированный доступ  
ПО – программное обеспечение  
СУБД – система управления базами данных