

Вестник

Томского государственного университета

ПРИЛОЖЕНИЕ

№ 1(II)

СЕНТЯБРЬ 2002

*Материалы
научных конференций,
симпозиумов, школ,
проводимых в ТГУ*



В Е С Т Н И К
ТОМСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

ОБЩЕНАУЧНЫЙ ПЕРИОДИЧЕСКИЙ ЖУРНАЛ

Свидетельства о регистрации: бумажный вариант № 018694, электронный вариант № 018693
выданы Госкомпечати РФ 14 апреля 1999 г.
ISSN: печатный вариант – 1561-7793; электронный вариант – 1561-803X
от 20 апреля 1999 г. Международного Центра ISSN (Париж)

№ 1 (II)

ПРИЛОЖЕНИЕ

Сентябрь 2002

ДОКЛАДЫ

IV Всероссийской конференции с международным участием
«Новые информационные технологии в исследовании сложных структур»
и Сибирской научной школы-семинара «Проблемы компьютерной безопасности»
(Томск, ТГУ, 10 – 13 сентября 2002 г.)

СОДЕРЖАНИЕ

Секция 1

ЛОГИКА И ДОКАЗАТЕЛЬСТВА

| | |
|--|----|
| Ануреев И.С. Система машинной поддержки доказательства: от тактикалов к генератору тактикалов | 4 |
| Москвитин А.А. Оценка длины убедительных доказательств в языках спецификаций задач | 8 |
| Москвитин А.А., Тимофеева М.К. Спецификация задач на основе трактовки естественного языка как формального..... | 14 |
| Shilov N.V., Garanina N. Model Checking Knowledge and Fixpoints | 20 |

Секция 2

ДИСКРЕТНЫЙ АНАЛИЗ

| | |
|--|----|
| Абросимов М.Б. Минимальные расширения графов, содержащих изолированные вершины..... | 24 |
| Андреева Л.Н., Оранов А.М. Алгоритм перечисления всех максимальных немонотонно допустимых подмно- жеств конечного множества | 30 |
| Евдокимов А.А., Левин А.А. Комбинаторные свойства и визуализация символьных последовательностей..... | 32 |
| Парватов Н.Г. О конечной порождаемости замкнутых классов функций многозначной логики | 34 |

Секция 3

ВЫЧИСЛИТЕЛЬНАЯ СЛОЖНОСТЬ

| | |
|--|----|
| Беспалов Д.В., Буранов Е.В., Семенов А.А., Черкашин Е.А. Логический анализ задачи 2-ФАКТОРИЗАЦИЯ | 38 |
| Быкова С.В. Квантовые вычисления и квантовый компьютер | 43 |
| Паршина Н.А. Полиномиальная выполнимость КНФ..... | 49 |

Секция 4

КРИПТОГРАФИЯ И ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

| | |
|--|----|
| Агибалов Г.П., Скутин А.А. Математическая модель безопасной корпоративной информационной системы..... | 53 |
| Агибалов Г.П., Скутин А.А. Технология разработки безопасных корпоративных информационных систем одно- го класса | 61 |
| Емельянов Г.В., Применко Э.А. Современные криптографические методы и средства защиты информации | 64 |
| Малюк А.А. Концепция информационной безопасности единой образовательной среды..... | 69 |
| Салий В.Н. Криптоанализ системы распределения ключей, основанной на полукольцах | 72 |
| Семенов А.А. Логический подход к анализу криптографических игр | 74 |
| Черемушкин А.В. Проблемы компьютерной криптографии | 80 |

Секция 5

ЛОГИЧЕСКИЙ СИНТЕЗ И ЛОГИЧЕСКИЙ АНАЛИЗ

| | |
|---|-----|
| Ветрова М.В. Минимальные детерминированные редукции недетерминированных автоматов..... | 87 |
| Кардаш С.Н. О реализации систем булевых функций матричными схемами с последовательными соединениями транзисторов..... | 92 |
| Люлькин А.Е. Логическое моделирование функционально-переключательных КМОП-структур..... | 96 |
| Оранов А.М. Метод построения дискретных схем на базе комплексных программируемых логических устройств | 102 |
| Панкратова И.А. Синтез двухкаскадных переключательных схем, реализующих функции на полурешётках | 108 |
| Поттосин Ю.В. Кодирование состояний асинхронного параллельного автомата кодами минимальной длины | 110 |
| Поттосин Ю.В., Томашев В.Ф., Шестаков Е.А. Обобщенное разложение Шеннона и его применение для мультиплексной декомпозиции булевых функций | 116 |
| Черемисинов Д.И. Применение языка XML в САПР дискретных устройств | 122 |
| Черемисинова Л.Д. Синтез многоуровневых сетей в базе ПЛИС на основе факторизации | 127 |
| Yevtushenko N., Villa T., Brayton R., Petrenko A., Sangiovanni-Vincentelli A. Solution of Synchronous Language Equations for Logic Synthesis..... | 132 |
| Steinbach B., Kempe G. Decomposition and Distributed Computing..... | 138 |

Секция 6

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

| | |
|---|-----|
| Александрова Т.В., Громаков Е.И. Нечеткая диагностическая система контроля качества производственного процесса | 143 |
| Кожеников С.Н., Козлов А.В., Мазуров Вл.Д., Первушин Д.Р. Экспертизы и отбор в принятии решений при условиях слабой формализуемости данных и знаний..... | 148 |
| Корниенко А.А., Корниенко А.В. Метод выявления экспертных диагностических знаний о поведении целеустремленных объектов | 154 |
| Лукин Н.А., Сафетдинов А.Р. Электронная библиотека рефератов E-Lib – базовые концепции, практические результаты и возможные перспективы | 157 |
| Микони С.В., Сорочкина М.И. Использование когнитивной графики в системе многокритериальной оценки объектов..... | 161 |
| Пудов С.Г. Обучение однородных клеточно-нелинейных сетей | 166 |
| Пушков С.Г. К общей теории нечетких систем: глобальные состояния, условия согласованности и линейность..... | 169 |
| Скрипин С.В. Многоуровневая работа со знаниями | 175 |
| Чистов В.П., Захарова Г.Б., Кононенко И.А., Титов В.Г. Экспертная система на основе нечёткой поведенческой логической модели технологического процесса доменного производства | 180 |
| Янковская А.Е., Гедике А.И., Аметов Р.В. Конструирование прикладных интеллектуальных систем на базе инструментального средства ИМСЛОГ-2002 | 185 |
| Яшин А.М., Гелеверя Т.Е., Руфанов А.В., Раков Г.Д. Один подход к построению систем управления знаниями | 190 |

Секция 7

ИНФОРМАЦИОННЫЕ СИСТЕМЫ В ЗДРАВООХРАНЕНИИ

| | |
|--|-----|
| Берестнева О.Г., Иванов В.Т., Иванкина Л.И., Муратова Е.А., Шаропин К.А. Информационная система мониторинга здоровья студентов | 196 |
| Евдокимов Д.А., Климина О.В., Никитина М.И. Автоматизация сбора медицинской статистической отчетности..... | 201 |
| Жучков Д.В., Кардашов Д.В., Никитина М.И. Структура и функции территориального хранилища медицинских данных..... | 206 |
| Исаев С.В., Исаева О.С., Ноженкова Л.Ф. Применение геоинформационной системы для анализа состояния здоровья населения | 212 |
| Янковская А.Е., Гергет О.М., Берестнева О.Г., Михалев Е.В. Подсистема логико-вероятностного принятия решения, опыт ее применения для оперативной оценки состояния новорожденных детей..... | 216 |

Секция 8

ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ

| | |
|---|-----|
| Бандман О.Л. Дискретно-непрерывные модели пространственной динамики | 222 |
| Климова О.В. Быстрые параллельные алгоритмы и рекурсивная псевдодвумерная декомпозиция свертки..... | 227 |
| Маркова В.П. Архитектура клеточного алгоритма деления | 232 |
| Медведев Ю.Г. Моделирование трехмерных потоков клеточными автоматами | 236 |
| Тимошевская Н.Е. О распараллеливании обхода дерева поиска..... | 241 |
| Tarkov M.S., Mun Y., Choi J., and Choi H. Parallel Synthesis of Fuzzy Kohonen Clustering Network for Image Segmentation | 245 |

Секция 9

РЕШЕНИЕ ЛОГИЧЕСКИХ УРАВНЕНИЙ

| | |
|---|-----|
| Бибило П.Н. Декомпозиция булевых функций на основе решения логических уравнений..... | 250 |
| Жарикова С. Оптимизация элементов цифровых схем посредством решения систем автоматных уравнений..... | 255 |
| Закревский А.Д., Василькова И.В. Решение больших систем логических уравнений: метод минимизации дерева поиска | 260 |
| Спицына Н.В., Евтушенко Н.В., Петренко А.Ф. Решение автоматного уравнения для обобщенной операции композиции | 265 |
| Штарнова О.В. К декомпозиции сетей Петри..... | 270 |
| Kirienko N.A. The C++ Classes for a System of Logic Equations..... | 274 |

Секция 10

ДИАГНОСТИКА

| | |
|---|-----|
| Бузанов В.А. О применении специальных операций в задачах диагностики..... | 278 |
| Кузнецов О.Ю., Шаршунов С.Г. Подход к проблеме функционального тестирования управляющего оборудования RISC-микропроцессоров..... | 284 |
| Матросова А.Ю., Седов Ю.В. О свойствах неисправностей, порожденных многоуровневыми методами синтеза, примененными к частично монотонным системам булевых функций..... | 287 |
| Новиков А.С., Шаршунов С.Г. Разработка алгоритмов функционального тестирования ОЗУ ЭВМ..... | 292 |
| Островский В.И., Левин И.С., Останин С.А. Обнаружение однонаправленных ошибок в управляющих устройствах на основе сокращенного m/n -кода | 296 |
| Cavalli A.R., Prokopenko S. Passive Testing Using Configuration Distinguishability..... | 301 |
| Goloubeva O., Reorda Sonza M., Violante M. Behavioral-Level Fault Models Comparison: an Experimental Approach..... | 305 |
| Trenkaev V., Kim M. Test Derivation Method for a System of Communicating FSMS with Respect to Output Faults..... | 310 |
| Ubar R., Raik J., Ivask E., Brik M. Test Cover Calculation in Digital Systems with Word-Level Decision Diagrams..... | 315 |

Секция 11

ОТКАЗОУСТОЙЧИВОСТЬ

| | |
|--|-----|
| Воробьев В.А., Лаходынова Н.В. Отказоустойчивость процессорных матриц: консенсус-парадигма..... | 320 |
| Гришин В.Ю., Лобанов А.В., Сиренко В.Г. Взаимное информационное согласование на основе глобального синдрома с обнаружением и идентификацией кратных враждебных неисправностей..... | 325 |
| Ерёмина Н.Л. Корректность алгоритма адресации для реконфигурации процессорной матрицы путем диагностического захвата..... | 330 |
| Каравай М.Ф. Об отказоустойчивом вложении произвольных гамильтоновых графов | 334 |
| Лаходынова Н.В. Об эффективности алгоритмов реконфигурации процессорной матрицы на основе консенсуса | 339 |
| Пархоменко П.П. Максимальные циклы в неисправных гиперкубовых структурах МВС..... | 343 |
| Frenkel S.L. About Probabilistic Models of Performance Evaluation..... | 345 |

Секция 12

ИНФОРМАТИКА

| | |
|--|-----|
| Браславский П.И. Метапоисковая машина для поиска специализированной научной информации в Интернете: структура и функции..... | 353 |
| Dorofeeva M., Koufareva I. FSM-based Strategies of Test Suite Minimization | 357 |
| Змеев О.А., Моисеев А.Н., Новиков Д.В. К вопросу проектирования уровня хранения в виде ООРБД..... | 363 |
| Кислицкий А.С., Куфарева И.Б. Сравнительный анализ методов поиска записей в плоском файле..... | 367 |
| Ковалев И.В., Алимханов А.М. Модели и алгоритмы оптимизации состава модульного программного обеспечения систем управления автономными объектами..... | 372 |
| Петренко С.В., Сибирякова В.А. Редактор и интерпретатор кадров обучения – РИКО..... | 375 |
| Руфанов А.В. Архитектура КИС на основе объектно-ориентированного ядра | 380 |
| Шакиров Р.Н. Оценка эффективности динамических массивов с автоматической проверкой индекса..... | 383 |
| РЕФЕРАТЫ СТАТЕЙ | 389 |

К ВОПРОСУ ПРОЕКТИРОВАНИЯ УРОВНЯ ХРАНЕНИЯ В ВИДЕ ООРБД

О.А. Змеев, А.Н. Моисеев, Д.В. Новиков

Филиал Кемеровского государственного университета, г. Анжеро-Судженск

E-mail: dim@asf.ru

В работе исследованы проблемы, возникающие при проектировании уровня хранения данных в сложных информационных системах, реализованных с применением объектной технологии. Рассмотрены существующие подходы решения задачи хранения объектов в гибридной базе данных. Предложен метод, позволяющий объединить технологию распределенного хранения объектов с шаблоном, использующим механизм метаданных.

Ключевые слова: *способы хранения данных объекта в реляционной базе данных, объектно-ориентированное проектирование.*

В настоящее время основной концепцией, принятой при анализе и проектировании сложных информационных систем, является объектно-ориентированная, основанная на компонентной архитектуре парадигма программирования. Существует масса причин, которые позволяют обосновать, почему объектно-ориентированный подход завоевал такие прочные позиции в программировании. Перечислим только некоторые из них:

- объектно-ориентированная программа состоит из независимых друг от друга модулей. Их можно повторно использовать в последующих проектах, что существенно сокращает время и стоимость новых разработок;
- пока открытый интерфейс некоторого класса или компонента остается неизменным, его внутреннюю реализацию можно менять как угодно. Это позволяет ускорить процесс модификации программ и сделать архитектуру системы более открытой;
- в объектно-ориентированной программе пользовательский интерфейс отделен от бизнес-правил, и эти два важных аспекта программной системы можно изменять независимо друг от друга;
- наследование вносит в программу логическую структуру, связывая классы между собой по принципу «от общего к частному». Такой программный продукт легче понять, проще сопровождать и поддерживать.

Несмотря на законченность объектного подхода к анализу и проектированию информационных систем, вопрос о хранении данных в таких системах в настоящее время считается открытым. На сегодняшний день существует два основных способа организации хранения данных: реляционный и объектно-ориентированный. В литературе, посвященной этому вопросу, высказывается масса диаметрально противоположных мнений относительно модели хранения данных [1–8], начиная с идеи о практической идентичности объектной и реляционной моделей, требующей лишь незначительного расширения одной из них, и кончая явным противопоставлением, ведущим к выводу о невозможности их объединения.

Цель настоящей работы – показать, что эти концепции абсолютно не противоречат друг другу и не требуют каких-либо изменений для того, чтобы использоваться при разработке программной системы, обладающей всеми свойствами как объектных, так и реляционных моделей, и дать ряд практических рекомендаций по реализации подобного рода системы. Подробное теоретическое обоснование вышесказанного приведено в [9]. Не вдаваясь в подробности, можно сказать, что недостатки каждой модели неразрывно связаны с ее преимуществами и фактически противоположны друг другу. Реляционные системы (R-системы), в основном, критикуются за отсутствие гибкости, являющейся следствием формальности (а следовательно, строгости и стабильности), а объектные (O-системы) – за отсутствие формальности, являющейся следствием гибкости.

Мы основываемся на следующем предположении о практической завершенности как реляционной, так и объектной концепций. Эта идея основывается на следующих утверждениях:

- Один и тот же набор данных может одновременно описываться несколькими разными моделями.
- Изначально, реляционная и объектная модели – разные модели, более того, цели построения этих моделей тоже различны.
- Наконец, напомним известный результат о том, что структуру любой сложности можно нормализовать.

Рассмотрим основные свойства, присущие каждой из моделей хранения данных, подробнее.

Реляционная модель базируется на понятии «таблица». Таблица определяется для каждой сущности, моделируемой приложением. Таблица – единственный тип данных, который может определить приложение.

Информационная модель приложения преобразуется в форму таблиц. Но операции, которые могут быть выполнены над таблицей, определяются реляционной моделью. Между таблицами можно установить динамические связи, задав условие соединения, определяющее ограничения, которым должны отвечать столбцы этих таблиц. Представление и функционирование реляционной системы типов полностью сконцентрированы на сервере базы данных. С точки зрения реляционной базы данных существуют одна модель, одна система типов и один язык, который «определяет» условия работы. Приложения должны признавать эту модель и обеспечить преобразования между их представлением и представлением базы данных.

Приложение в реляционной среде обычно пишется на языке, который представляет быстродействующую среду макетирования для построения интерфейса с базой данных на основе набора форм. Приложение и реляционная база данных имеют отдельные системы типов. При этом типы данных и возможные действия содержат существенные отличия. Интерфейс между значениями типов столбцов и значениями встроенных типов языка программирования обеспечивается со стороны базы данных.

Архитектура реляционной базы данных не влияет на представление данных в приложении или управление ими. Приложение несет полную ответственность за создание всех экземпляров объектов и отслеживание всех модификаций их атрибутов. При получении доступа к объекту приложение должно быть уверено, что соответствующие данные не были еще модифицированы.

Поскольку система типов реляционной базы данных и среда выполнения полностью независимы от языка программирования приложения, реляционная база данных может лучше поддерживать несколько языков разработки приложений. Более того, в настоящий момент существует достаточно широкий выбор коммерческих СУБД.

Реляционные таблицы могут соответствовать объектам объектной модели, но на этом уровне интерфейс не предоставляется. В результате, объектная модель должна иметь два представления: схему реляционной базы данных и классов приложения. При разработке приложения между этими моделями должно быть определено и реализовано некоторое преобразование.

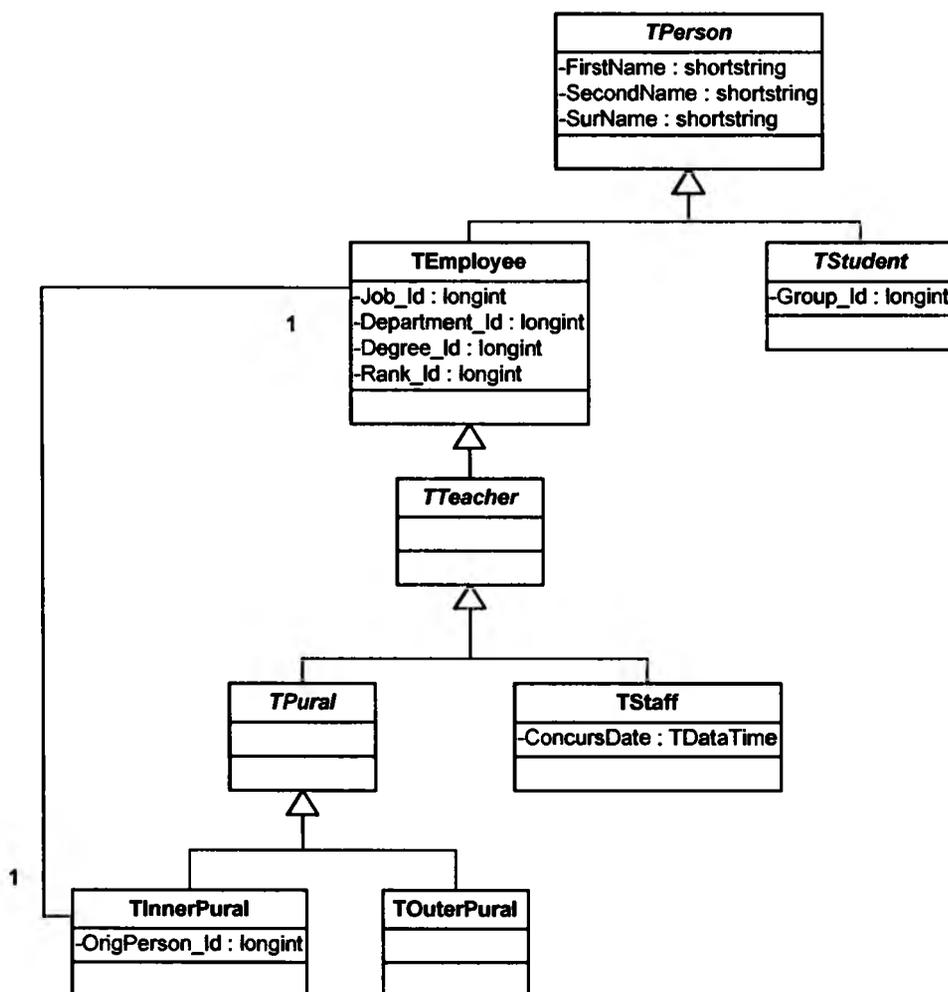


Рис. 1

Появление объектно-ориентированной парадигмы внесло фундаментальные изменения во взгляды на данные и процедуры их обработки – процедуры обработки сущностей и данные хранятся вместе. Такое совместное хранение считается шагом вперед в методах управления данными. Сущности становятся замкнутыми единицами, которые очень легко можно использовать повторно и перемещать в новое место. Теперь поведение сущности не связано с прикладной программой, а является частью самой сущности. Объектная модель поддерживает связи «многие-ко-многим». Объектно-ориентированные базы являются навигационными: доступ к данным производится с помощью связей, хранящихся внутри самих данных, при этом есть один недостаток – формулирование незапланированных запросов к объектно-ориентированной базе данных осуществляется достаточно тяжело.

Существует несколько способов хранения данных классов в реляционной базе данных, первый из которых предполагает, что для хранения каждого из них предоставляется отдельная реляционная таблица. Второй способ предполагает распределенное многоуровневое хранение объектов с выделением хэш-таблицы [9]. Рассмотрим достоинства и недостатки этих методов хранения.

Первый способ приводит к разработке довольно гибкой системы, программная реализация которой не испытывает проблем при расширении системы, но снижает степень нормализации и приводит к увеличению объемов базы данных за счет дублирования совпадающих атрибутов в различных объектах. Второй способ отличается более высокой степенью нормализации, но при таком способе хранения мы получаем жесткую структуру, и изменение иерархии классов приводит к серьезным проблемам с точки зрения программной реализации и хранения данных.

Разрабатывая информационную систему «Факультет», мы пришли к выводу, что ни один из описанных выше способов хранения в достаточной степени не подходит для решения задачи хранения данных в сложной системе. Количество классов этой системы достаточно велико, а вероятность расширения системы достаточно высока. Рассмотрим лишь небольшую часть иерархии классов, касающуюся понятия «Человек», и способы организации хранения этих данных.

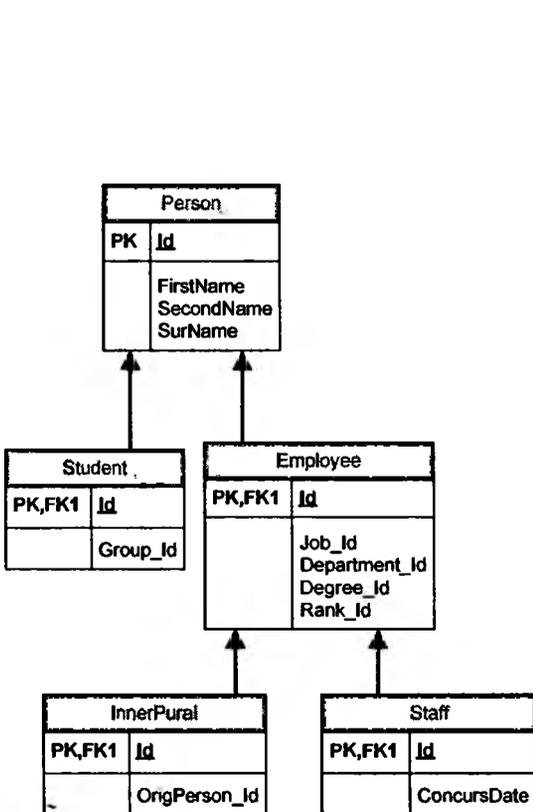


Рис. 2

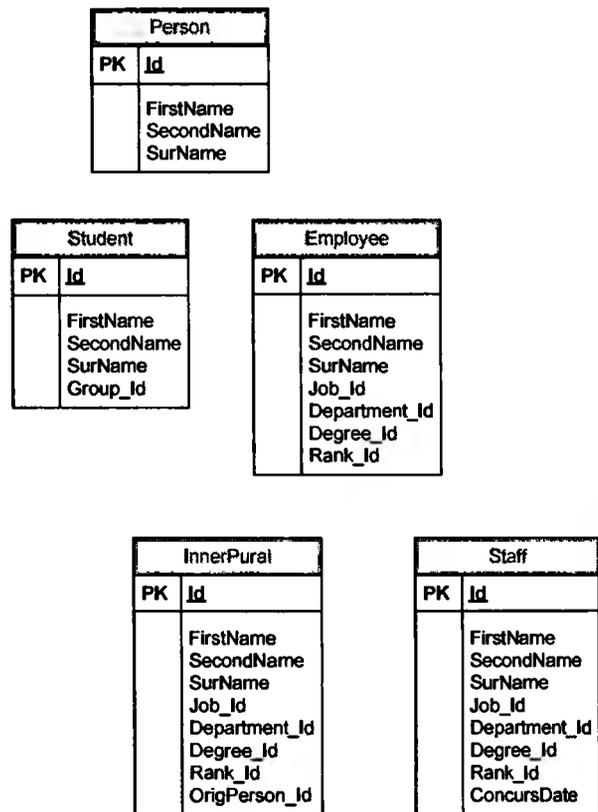


Рис. 3

Как видно, в первом случае наблюдается дублирование данных, в частности дублирование таких атрибутов, как *FirstName*, *SecondName* и *Surname* (Фамилия, Имя, Отчество), что даже в этом случае приводит к достаточно большому увеличению объемов данных. Во втором случае с точки зрения нормализации все хорошо, но к каким проблемам приводит расширение иерархии классов? Добавление некоторого класса в иерархию приводит к изменению структуры таблиц, так как некоторые атрибуты приходится выносить на предыдущий уровень иерархии и в дальнейшем перестраивать ветвь. Это довольно проблематично, а на

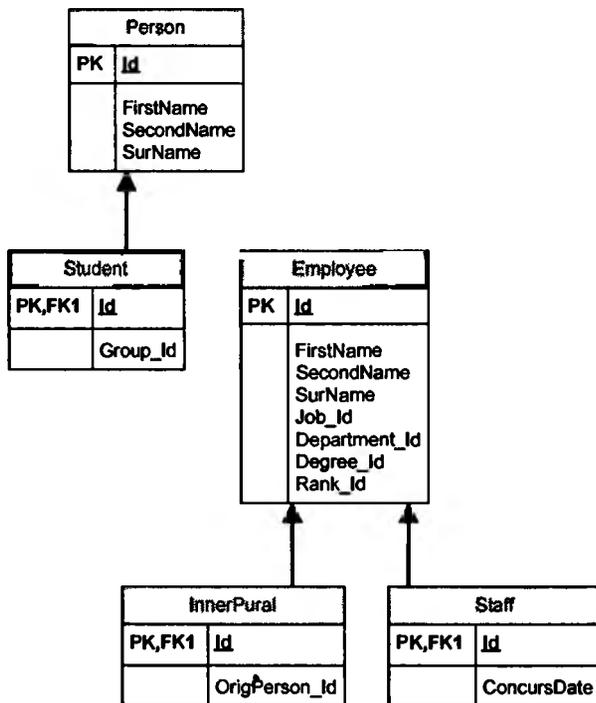


Рис. 4

функционирующей системе расширить иерархию еще проблематичнее, так как в данном случае приходится перемещать не только атрибуты, но и их значения. Причем, вам приведена лишь крошечная часть классов рассматриваемой системы.

В качестве выхода из сложившейся ситуации мы предлагаем некоторый компромисс. При проектировании классов уровня хранения выделяются некоторые группы классов, которые определяются по принципу схожести функциональных обязанностей, причем такое разделение производится только для уровня хранения данных, в предметной области все аспекты реализации системы, связанные с наследованием, остаются классическими. Внутри выделенных групп используется вариант плоского распределенного хранения данных, а для описания доступа к самой группе – видоизмененный шаблон, использующий механизм метаданных. Например, при проектировании уже рассмотренной иерархии классов можно изначально разделить сущности *Student* и *Employer* (Студенты и Сотрудники), что, конечно, приведет к некоторому дублированию данных, но сделает систему более гибкой при внесении в нее изменений на уровне хранения. Более того, такой подход делает набор классов более понятным на этапе анализа и проектирования информационной системы, он совпадает с понятиями, кото-

рые используют для описания системы эксперты или системные аналитики, при этом все преимущества объектно-ориентированного подхода сохраняются.

Реализация такого контура уровня хранения позволяет получить набор классов с высокой степенью зацепления, взаимодействующих между собой и обеспечивающих обслуживание ядра, включающий конкретные классы, которые определяют интерфейс между объектами. Этот набор требует определения пользователем подклассов для существующих классов контура, их использования, настройки и расширения служб, имеет абстрактные классы, которые могут содержать как абстрактные, так и конкретные методы.

ЛИТЕРАТУРА

1. Дейт К. Введение в базы данных. Изд. 6-е. – Киев: Диалектика, 1998.
2. Дарвин Х., Дэйт К. Третий манифест // СУБД. – 1996. – № 1.
3. Аткинсон М., Бансилон Ф., ДеВитт Д. и др. Манифест систем объектно-ориентированных баз данных // СУБД. – 1995. – № 4.
4. Системы баз данных третьего поколения: Манифест // СУБД. – 1995. – № 2.
5. Пржиялковский В.В. Новые одежды знакомых СУБД: Объектная реальность, данная нам // СУБД. – 1997. – № 4.
6. Базы данных: достижения и перспективы на пороге 21-го столетия / Под ред. А. Зильбершатца, М. Стоунбрейкера и Д. Ульмана // СУБД. – 1996. – № 3.
7. Вон К. Технология объектно-ориентированных баз данных // Открытые системы. – 1994. – № 4.
8. Стоунбрейкер М. Объектно-реляционные системы баз данных // Открытые системы. – 1994. – № 4.
9. Ларман К. Применение UML и шаблонов проектирования: Издательский дом «Вильямс», 2001. – 496 с.