



ИНСТИТУТ ТРАНСПОРТА И СВЯЗИ

ПЕТУХОВА Наталья Юрьевна

**ТЕМПОРАЛЬНЫЕ МОДЕЛИ ДАННЫХ
В ИНФОРМАЦИОННЫХ СИСТЕМАХ
НА ЖЕЛЕЗНОДОРОЖНОМ ТРАНСПОРТЕ**

**АВТОРЕФЕРАТ ДИССЕРТАЦИОННОЙ РАБОТЫ
на соискание учёной степени доктора инженерных наук
(Dr.sc.ing.)**

Научная область “Транспорт”
направление “Транспорт и логистика”

Научный руководитель
Dr.habil.sc.ing., профессор
Копытов Евгений Александрович

РИГА – 2010

UDK 004.6:656.2

П 314

Transporta un sakaru institūts

Институт транспорта и связи

Петухова Н. Ю.

П 314 Темпоральные модели данных в информационных системах на железнодорожном транспорте. Автореферат диссертационной работы. Рига: Институт транспорта и связи, 2010. 58 с.

ISBN

© Петухова Н. Ю., 2010

© Институт транспорта и связи, 2010

**ДИССЕРТАЦИОННАЯ РАБОТА ПРЕДСТАВЛЕНА В ИНСТИТУТ
ТРАНСПОРТА И СВЯЗИ НА СОИСКАНИЕ УЧЁНОЙ СТЕПЕНИ
ДОКТОРА ИНЖЕНЕРНЫХ НАУК (DR.SC.ING.)**

ОФИЦИАЛЬНЫЕ РЕЦЕНЗЕНТЫ:

Dr.habil.sc.ing., профессор Л. П. Новицкий,
Рижский Технический Университет, Латвия

Dr.habil.sc.ing., профессор Г.И. Пранявичус,
Каунасский Технологический Университет, Литва

Dr.sc.ing., профессор Б. Ф. Мишнёв,
Институт транспорта и связи, Латвия

Защита диссертационной работы состоится 30 марта 2011 г. в 15:00 в промоционном совете Института транспорта и связи по адресу: Латвия, г. Рига, ул. Ломоносова 1, ауд. 130, тел. (+371) 7100617, факс: (+371) 7100535.

ПОДТВЕРЖДЕНИЕ

Я подтверждаю, что выполнила данную диссертационную работу, которая представлена на рассмотрение в Институт транспорта и связи на соискание учёной степени доктора инженерных наук (Dr.sc.ing.). Данная диссертационная работа ранее не представлялась на рассмотрение в другие промоционные советы.

___ 2010 г. Н. Ю. Петухова

Данная диссертационная работа написана на английском языке, состоит из 4 глав и включает 80 рисунков, 50 формул, 12 таблиц и 4 приложения, составляя в общем 164 страницы. Список используемой литературы включает 134 источника.

АННОТАЦИЯ

Диссертационная работа Натальи Юрьевны Петуховой “Темпоральные модели данных в информационных системах на железнодорожном транспорте”. Научный руководитель хабилитированный доктор инженерных наук, профессор Евгений Александрович Копытов.

В настоящей работе представлены результаты исследований по проблемам использования темпоральных данных в информационных системах (ИС) на железнодорожном транспорте, проведённые автором в период с 2001 по 2010 гг. В качестве объекта приложения результатов исследований выбран комплекс информационных систем, используемых в ГАО “Латвийская железная дорога” (ЛЖД). Вместе с тем, полученные результаты носят универсальный характер и могут быть применены другими транспортными предприятиями. В работе исследована специфика темпоральных ИС на железнодорожном транспорте; выявлены и классифицированы основные задачи ИС, связанные с учётом и обработкой темпоральных данных на ЛЖД; определены проблемы, возникающие в процессе эксплуатации и проектирования этих систем.

Рассмотрены вопросы представления темпоральных объектов в базах данных, обеспечения их целостности и взаимодействия с ними. Исследованы особенности реализации временных принципов в рамках реляционной модели данных и предложена модель темпоральных данных с абстрактным идентификатором объекта. Исследована возможность ее совместимости с действующими ИС. Введена новая темпоральная форма Historical-Overlapped, которая учитывает особый режим смены версий темпорального объекта, свойственный некоторым процессам на железнодорожном транспорте. Разработана модель темпорального объекта, обладающего периодичностью и зависимостью от календаря. Исследована задача определения актуальной версии темпорального объекта. Для ее решения предложены два альтернативных метода: метод логических правил и метод темпоральных элементов. Разработан метод разграничения доступа к данным реляционных систем на уровне строк отношения, который решает проблему управления доступом к темпоральным данным различных временных периодов. Предложен метод трансформации интервальных данных, решающий проблему сложных вычислений при выполнении аналитических запросов. Исследованы вопросы учёта темпоральной составляющей в процессах управления хранилищами данных.

Приведены примеры применения полученных результатов при решении ряда практических задач. Результаты исследований применены в информационных системах, эксплуатируемых на Латвийской железной дороге.

СОДЕРЖАНИЕ

1. АКТУАЛЬНОСТЬ РАБОТЫ.....	6
2. СТЕПЕНЬ РАЗРАБОТАННОСТИ ТЕМЫ	7
3. ЦЕЛЬ И ЗАДАЧИ ИССЛЕДОВАНИЯ	10
4. МЕТОДОЛОГИЯ И МЕТОДЫ ИССЛЕДОВАНИЯ.....	10
5. НАУЧНАЯ НОВИЗНА РАБОТЫ.....	11
6. ПРАКТИЧЕСКАЯ ЦЕННОСТЬ РАБОТЫ И РЕАЛИЗАЦИЯ.....	11
7. АПРОБАЦИЯ РАБОТЫ	12
8. ПУБЛИКАЦИИ	13
9. СТРУКТУРА РАБОТЫ	14
10. ОПИСАНИЕ ОСНОВНЫХ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ	15
10.1. Модели и методы для работы с темпоральными данными в ИС железнодорожного транспорта	15
10.2. Модель темпоральных данных с абстрактным идентификатором объекта.....	21
10.3. Модель темпорального отношения с пересекающимися жизненными отрезками Historical-Overlapped	24
10.4. Реализация временной логики в рамках реляционной модели	27
10.5. Модель периодичности в задачах железнодорожного расписания	32
10.6. Методы определения актуальной версии темпорального объекта.....	37
10.7. Метод контроля доступа к темпоральным данным на уровне кортежей отношения.....	44
10.8. Метод трансформации интервальных данных.....	46
10.9. Применение результатов исследования в системах хранилищ данных	48
10.10. Применение результатов исследования в интерактивной информативно-справочной системе расписания пассажирских поездов	51
ЗАКЛЮЧЕНИЕ	54
СПИСОК ОПУБЛИКОВАННЫХ АВТОРОМ РАБОТ	55
Статьи	55
Тезисы	57
Нормативные документы и концепции	58

1. АКТУАЛЬНОСТЬ РАБОТЫ

Большинство процессов на транспорте связано со временем. Временную основу имеют задачи планирования и прогнозирования деятельности транспортного предприятия, практически все аналитические и оперативные задачи требуют использования в информационных системах (ИС) данных, измеренных в различные моменты времени. Многие из этих данных, называемых *темпоральными данными*, являются актуальными только на определенном временном отрезке, они связаны с определенными датами (временными метками) или временными интервалами. Правильная организация темпоральных данных в базах данных (БД) и средств доступа к ним во многом определяет эффективность функционирования ИС транспортного предприятия.

В каждом виде транспорта имеются свои дополнительные специфические особенности, которые необходимо учитывать при построении ИС, использующих темпоральные данные. В частности, железнодорожный транспорт отличается более жестким расписанием, обусловленным физическими ограничениями транспортной сети, наличием множества изменений в графике движения поездов, большим количеством промежуточных станций, взаимосвязью расписаний пассажирских и грузовых поездов. Естественно, построение баз данных, характеризующих состояние системы железнодорожного транспорта в любой момент предыстории, является исключительно сложной и наукоёмкой задачей. Как результат, несмотря на постоянное развитие комплекса ИС на железнодорожной дороге, достаточно часто пользователи систем испытывают серьёзные трудности с получением необходимых данных, характеризующих состояние хранимых объектов в различные моменты времени. Как правило, в базах данных ИС хранятся только выборочные значения показателей, а часто и просто их усредненные по времени значения. В связи с этим данные приходится восстанавливать из различных источников, производить аппроксимацию, экстраполяцию, что вносит в используемый набор данных массу допущений, осреднений и погрешностей.

По мнению автора, такое положение с данными связано с применением в ИС технологии классических реляционных баз данных, которые не ориентированы на хранение всех изменений объектов БД и не поддерживают их многоверсионность. При этом темпоральные принципы в этих системах используются неэффективно. Это проявляется в следующих недостатках ИС:

- несовершенство и низкая описательная составляющая моделей темпоральных объектов;
- неэкономное использование дискового пространства;
- низкая эффективность взаимодействия с темпоральными объектами;
- плохая расширяемость БД;
- высокая трудоемкость разработки БД;
- сложность поддержки целостности данных;
- сложность организации доступа к данным;
- проблемы в обеспечении конфиденциальности данных.

В аналитических ИС этот список можно дополнить такими недостатками, как:

- отсутствие аппарата эффективной обработки темпоральных данных, вследствие чего огромный потенциал информации, скрытый в большом объеме накопленных данных, используется не полностью;
- несовершенство процессов подготовки данных к анализу, которые не ориентированы на темпоральный характер данных и имеют проблемы, связанные с обеспечением целостности и доступности данных.

Сдерживающим фактором развития технологии темпоральных БД (ТБД) многие годы являлось высокое требование к ресурсам по поддержке многоверсионности объектов хранения. Но, в связи со стремительным развитием вычислительной техники, он стал отступать на второй план. Однако, существует ещё ряд других факторов, затрудняющих использование ТБД на таких крупных предприятиях, как железная дорога, среди них следующие: отсутствие единого стандарта обработки темпоральных данных; специфика темпоральных объектов систем предприятия; отсутствие навыков по работе с темпоральными данными у большинства разработчиков ИС; повышенные требования к целостности, доступности и конфиденциальности данных.

Перечисленные обстоятельства свидетельствуют об актуальности работ, направленных на повышение эффективности использования темпоральных данных в информационных системах на железнодорожном транспорте, что и явилось основанием для проведения настоящего исследования.

2. СТЕПЕНЬ РАЗРАБОТАННОСТИ ТЕМЫ

Управление темпоральными данными информационных систем на железнодорожном транспорте является комплексной задачей, исследование которой проводится по следующим направлениям:

1. Построение и функционирование ИС на железнодорожном транспорте;
2. Проектирование темпоральных баз данных и ИС: стандарты, методы, модели, подходы;
3. Оперативная и аналитическая обработка темпоральных данных;
4. Обеспечение информационной безопасности темпоральных данных.

В области *построения и функционирования ИС* на железнодорожном транспорте необходимо отметить труды российских ученых Э.К. Лецкого, В.И. Панкратова, В.А. Шарова, Ю.Ф. Мухопода и др. В их исследованиях рассмотрены ИС железных дорог разных стран, показана сложность и специфичность работы железной дороги и её информатизации, предложен ряд моделей функционирования на макро- и микро- уровнях. Отмечая важность учёта фактора времени, авторы все же не предлагают специальных решений в этой области, ориентируясь на классические подходы.

Среди исследований, выполненных применительно к ИС Латвийской железной дороги, следует отметить промоционную работу В.В. Демидова (2006 г.), направленную на повышение эффективности систем поддержки принятия решений на железной дороге путём совершенствования структуры и процессов функционирования системы информационного обеспечения. В

данной работе предложены модели, описывающие топологию транспортной сети железнодорожной дороги и процессы пассажирских перевозок, разработаны модели и методы представления и преобразования данных в рассматриваемых информационно-аналитических системах, в том числе виртуальные вероятностные модели данных, метод иерархически связанных представлений, концептуальные модели прогнозирования железнодорожных перевозок и др. Говоря об исследовании проблемы применения темпоральных данных, отметим решение ряда задач, связанных с накоплением и хранением истории изменений состояний объектов, а также предоставлением средств доступа к ним в любой момент времени. Заслуживает внимания разработка принципов построения ТБД и реализации некоторых элементов темпоральной логики (здесь В.Демидов и автор настоящей диссертации работали совместно [3, 5, 12]). Вместе с тем многие проблемы темпоральных систем, возникающие при разработке и функционировании ИС на транспорте, В.Демидовым не были решены или вообще не рассматривались, в первую очередь следует назвать учёт периодичности событий и использования особого календаря, вопросы поддержки целостности и конфиденциальности темпоральных данных.

Актуальные проблемы и достижения в области ТБД обобщены в совместной работе С.Д.Кузнецова и Б.Костенко. Среди фундаментальных работ, посвященных *темпоральным технологиям*, выделяются работы ученых К. Дейта (C. J. Date), Х. Дарвена (Hugh Darwen), Н. Лоренцоца (Nikos A. Lorentzos), К. Йенсена (Christian S. Jensen). Их совместный труд “Temporal Data and the Relational Model” рассматривает вопросы хранения и взаимодействия с темпоральными данными в рамках реляционной модели. Однако авторы в этой работе концентрируются на общих вопросах построения ТБД и использования языка Tutorial D. Так, практически не рассмотрены вопросы, связанные с учётом периодичности и особого календаря, что является исключительно важным в транспортных ИС, в первую очередь в задачах расписания движения транспортных средств.

Одним из наиболее обширных исследований в области ТБД является комплекс работ К. Йенсена (Christian S. Jensen), выполненных в период с 1991 по 1999 год в соавторстве с известными учеными разных стран, такими как: Д. Бэр (John Bair), Р. Блюй (Rasa Bliuj), М. Болен (Michael H. Böhlen), Р. Буссато (Renato Busatto), Д. Клиффорд (James Clifford), К. Дайрсон (Curtis E. Dyreson), Ф. Гранди (Fabio Grandi), Х. Греггерсен (Heidi Gregersen), С. Хсу (Suchen Hsu), Т. Исакович (Tomáz Isakowitz), Т. Клифф Леунг (T. Y. Cliff Leung), Л. Марк (Leo Mark), С. Шалтенис (Simonas Šaltenis), Г. Сливинскас (Giedrius Slivinskas), Я. Скит (Janne Skyt), Р. Снодграсс (Richard T. Snodgrass), М. Су (Michael D. Soo), А. Штайнер (Andreas Steiner), К. Торп (Kristian Torp). В 2000 году это собрание работ опубликовано в виде диссертации объёмом более 1300 страниц. Большая часть перечисленных ученых участвовала в комиссии по разработке спецификации языка запросов к темпоральным данным TSQL2 и стандарта SQL-3. Изучение TSQL2 и SQL-3 необходимо при разработке методов и моделей для работы с темпоральными данными.

Следует отметить огромный вклад в развитие технологии ТБД международного центра TimeCenter, занимающегося поддержкой темпоральных приложений на традиционных и появляющихся технологиях систем управления базами данных (СУБД), в состав которого входят ведущие ученые в этой области. Среди трудов TimeCenter выделяются работы П. Терензиани (Paolo Terenziani) и его коллег, особое внимание в которых уделено обработке временных ограничений на события, повторяющиеся с течением времени (так называемая "периодичность"). В них исследуется "символьный" язык высокого уровня для представления периодичности, который более ориентирован на пользователя, чем язык математических формул. В этой же работе рассматривается темпоральная реляционная модель, которая поддерживает определяемую пользователем "символьную" периодичность (например, "второй понедельник каждого месяца"). Определяется темпоральное дополнение к стандартным операторам реляционной алгебры, вводятся новые темпоральные операторы и функции. Предлагается временная алгебра, являющаяся непротиворечивым расширением классической (не временной) алгебры.

Реализация *оперативной и аналитической обработки темпоральных данных* базируется на исследованиях в области интеллектуальной обработки данных и систем принятия решений, а также теории хранилищ данных. Большой вклад в формирование основ многомерного представления данных и классических подходов к отслеживанию изменений в измерениях внесли работы Б. Инмона (B. Inmon), Р. Кимболла (R. Kimball), Н. Пендса (N. Pendse), Т. Педерсена (T. Pedersen).

Исследования, относимые к последнему блоку, связаны непосредственно с проблемами *информационной безопасности*. Здесь рассматриваются вопросы, посвященные дискреционной и мандатной моделям безопасности. Отдельные аспекты применения этих моделей в системах баз данных нашли свое отражение в исследованиях Н. Юкица (N. Jukic), С. Врбского (S.V. Vrbsky), Д. Деннинга (D.E. Denning), С. Яждоиа (S. Jajodia), Р. Санду (R. Sandhu), С. Осборна (S. Osborn). Общая концепция информационной безопасности отражена в стандартах: "Критерии определения безопасности компьютерных систем"¹, "Гармонизированные критерии европейских стран"², "Общие критерии оценки безопасности информационных технологий"³.

Высоко оценивая уровень вышеупомянутых работ, в то же время необходимо отметить, что с позиций выбранной темы они не обеспечивают решения многих проблем, возникающих при использовании темпоральных данных в ИС на железнодорожном транспорте. Из существующих исследований, проводимых на железной дороге, можно отметить ряд работ, посвященных узким вопросам оптимизации процессов, связанных со временем: например, задаче оптимизации алгоритма построения расписания поездов. Однако комплексных исследований на тему организации и использования

¹ Trusted Computer System Evaluation Criteria, TCSEC, 1985

² Information Technology Security Evaluation Criteria, ITSEC, 1991

³ Common Criteria for Information Technology Security Evaluation, ISO/IEC 15408, 2009

темпоральных данных с учётом специфики транспортных процессов не проводилось. В связи с этим можно утверждать, что проблема, рассматриваемая в настоящей работе, в предложенной постановке решается впервые.

3. ЦЕЛЬ И ЗАДАЧИ ИССЛЕДОВАНИЯ

Целью диссертационной работы является совершенствование процессов разработки и функционирования информационных систем на железнодорожном транспорте путём применения технологий темпоральных баз данных с детальной проработкой вопросов создания и использования темпоральных моделей данных и методов доступа к данным.

В соответствии с поставленной целью автором решаются следующие **задачи**:

1. Исследование специфики построения темпоральных ИС на железнодорожном транспорте и выявление проблем, требующих решения.
2. Критический анализ исследований в области темпоральных БД и построения темпоральных информационных систем, выбор и обоснование существующих подходов для практического применения в ИС железнодорожного транспорта, формулирование направлений настоящего исследования.
3. Разработка темпоральных моделей данных и методов взаимодействия с темпоральными объектами с учётом специфики задач обработки данных в ИС на железнодорожном транспорте.
4. Разработка методов защиты темпоральных данных, обеспечивающих повышенные требования ИС железнодорожных предприятий к целостности, конфиденциальности и доступности данных.
5. Разработка расширения реляционных средств манипулирования данными, реализующих трансформацию темпоральных данных в аналитических системах.
6. Применение результатов исследования для решения практических задач в ИС на Латвийской железной дороге.

4. МЕТОДОЛОГИЯ И МЕТОДЫ ИССЛЕДОВАНИЯ

Диссертационные исследования базируются на:

- результатах моделирования процессов на железнодорожном транспорте;
- результатах научно-исследовательских работ, выполненных в Институте транспорта и связи и ГАО «Латвийская железная дорога», в которых автор участвовала лично;
- опыте автора, приобретённом при работе в различных проектах по информационным технологиям в области баз данных и Web систем в

качестве эксперта, архитектора, разработчика баз данных и приложений;

- материалах научных конференций, в которых автор принимала непосредственное участие;
- технической документации, научно-технической литературе и периодических изданиях, посвященных рассматриваемым в диссертации проблемам.

Методами исследования, использованными в работе, являются математические методы системного анализа, булевой алгебры, реляционного исчисления и теории множеств.

5. НАУЧНАЯ НОВИЗНА РАБОТЫ

Работа представляет собой комплексное исследование вопросов построения и функционирования темпоральных информационных систем на железнодорожном транспорте.

Автор выносит на защиту следующие научные результаты:

1. Предложены принципы построения темпоральных баз данных в реляционной среде, адаптированные к условиям функционирования ИС на железной дороге.
2. Разработаны оригинальные модели и методы для работы с темпоральными объектами:
 - модель темпоральных данных с абстрактным идентификатором объекта;
 - модель темпорального отношения с пересекающимися жизненными отрезками Historical-Overlapped;
 - модель темпорального объекта, обладающего периодичностью и особым календарём;
 - метод обеспечения временного окружения для пользователя;
 - методы определения актуальной версии темпорального объекта: метод логических правил и метод темпоральных элементов;
 - метод трансформации интервальных данных;
 - метод защиты темпоральных данных реляционных систем на уровне кортежа отношения.

6. ПРАКТИЧЕСКАЯ ЦЕННОСТЬ РАБОТЫ И РЕАЛИЗАЦИЯ

Практическая значимость работы заключается в разработке и применении темпоральных технологий на железнодорожном транспорте.

Представленные результаты позволяют:

- наделять базы данных действующих ИС железнодорожного транспорта темпоральными свойствами и обеспечить темпоральную восходящую совместимость (без необходимости модернизации действующего прикладного программного обеспечения);

- усовершенствовать представление в БД темпоральных объектов и методы взаимодействия с ними;
- повысить целостность, доступность и конфиденциальность темпоральных данных;
- усовершенствовать процессы трансформации и анализа темпоральных данных в хранилищах данных, используемых на железной дороге.

На основе полученных результатов при активном участии автора

- разработана Интерактивная информативно-справочная система расписания пассажирских поездов на Латвийской железной дороге;
- усовершенствована Централизованная система железнодорожных классификаторов и кодификаторов, используемая в различных информационных системах на Латвийской железной дороге;
- модернизирован ряд информационных подсистем на Латвийской железной дороге, в том числе Финансово-статистическая информационно-аналитическая система пассажирских перевозок, Система учёта периферии вычислительной техники и др.

7. АПРОБАЦИЯ РАБОТЫ

Результаты, полученные в ходе исследований, докладывались на 16 научных и научно-практических конференциях в Бельгии, Израиле, Латвии, Литве и России, в их числе:

1. VI международная конференция “**TransBaltica 2001**”, Рига, Латвия. 7-8 июня 2001 г.;
2. Научно-практическая и учебно-методическая конференция “**Наука и технология - шаг в будущее**”, Институт транспорта и связию. Рига, Латвия, 2-3 мая 2002 г.;
3. VII международная конференция “**TransBaltica 2002**”. Рига, Латвия. 12-14 июня, 2002 г.;
4. 43-я международная научная конференция Рижского Технического Университета, Рига, Латвия. 10-14 октября 2002 г.;
5. Международная конференция “Reliability and Statistics in Transportation and Communication” **RelStat’02**. Рига, Латвия, 17-18 октября 2002 г.;
6. Международная научно-техническая конференция, посвященная 80-летию гражданской авиации России “Гражданская авиация на современном этапе развития науки, техники и общества” **МГТУГА**. Москва, 17-18 апреля 2003 г.;
7. Международная конференция “Modelling and Simulation of Business Systems” **MOSIBUS 2003**, Вильнюс, Литва, 13-14 мая 2003 г.;
8. VI международная конференция “Computing Anticipatory Systems” **CASYS '03**. Льеж, Бельгия, 11-16 августа 2003 г.;
9. Международная конференция “Reliability and Statistics in Transportation and Communication” **RelStat’03**. Рига, Латвия. 16-17 октября 2003 г.;

10. VI международная Балтийская конференция “Databases and Information Systems” **DB&IS 2004**. Рига, Латвия. 6-9 июня 2004 г.;
11. Международная конференция “Reliability and Statistics in Transportation and Communication” **RealStat’05**. Рига, Латвия. 13-14 октября 2005 г.;
12. Научно-практическая и учебно-методическая конференция “**Наука и технология - шаг в будущее**”, Институт транспорта и связи. Рига, Латвия. 15-16 декабря 2006 г.;
13. Международная конференция “Reliability and Statistics in Transportation and Communication” **RelStat’07**. Рига, Латвия. 24-27 октября 2007 г.;
14. Международная конференция “Modelling of Business, Industrial and Transport Systems” **MBITS’08**. Рига, Латвия. 7-10 мая 2008 г.;
15. Международный симпозиум “Stochastic Models in Reliability Engineering, Life Science and Operations Management” **SMRLO’10**. Беэр-Шева, Израиль. 8-11 февраля 2010 г.;
16. 9-я международная конференция “Joint Conference on Knowledge-Based Software Engineering” **JCKBSE’10**. Каунас, Литва.

Работы автора в области совершенствования ИС на транспорте были отмечены:

- наградой за лучшую научную статью “*Принципы создания хранилищ данных в системах поддержки принятия решений на железнодорожном транспорте*” симпозиума "Social and Economic Systems, Model, Simulation, Agents, Decision Support". Шестая международная конференция "Computing Anticipatory Systems" **CASYS '03**. 11-16 августа 2003, г. Льеж, Бельгия;
- премией имени Карлиса Ирбитиса (стипендия докторанта) от Латвийской Академии наук, АО „Latvijas Gaisa Satiksme” и Латвийского фонда образования „Образованию, науке и культуре” за работу “*Повышение безопасности транспортных информационных систем*”, 2005 г.

8. ПУБЛИКАЦИИ

По результатам исследований опубликовано 27 научных работ [1-27], в том числе 17 научных статей и 10 тезисов научных докладов. В них рассмотрены проблемы создания информационных систем на транспортных предприятиях и задачи, связанные с проектированием и реализацией темпоральных баз данных. В работах особое внимание уделено методам обработки больших объемов данных, вопросам обеспечения целостности, доступности и конфиденциальности темпоральных данных.

Кроме того, при непосредственном участии автора разработаны 2 концепции развития информационной системы расписания на Латвийской железной дороге [28, 32] и два технических задания к системам: „Электронное резервирование в международных поездах” [29] и учёта периферии

вычислительной техники „APNIS” [31], а также технология системы „Электронное резервирование в международных поездах” [30].

9. СТРУКТУРА РАБОТЫ

Работа состоит из введения, четырёх глав, заключения и пяти приложений. Содержит 80 рисунков и 8 таблиц, составляя в общем 164 страницы. Список используемой литературы включает 134 наименования.

Во Введении рассматривается актуальность темы диссертационной работы, формулируются цель и задачи исследования, показывается новизна и практическая ценность полученных результатов, даётся краткое содержание.

В Главе 1 описываются особенности и специфика ИС на железнодорожном транспорте, показывается роль времени и темпоральных данных в этих системах. Исследуются эволюция и проблемы использования темпоральных данных в ИС. Выявляются и классифицируются задачи ИС, связанные с учётом и обработкой темпоральных данных. Производится постановка задачи исследования.

В Главе 2 рассматриваются базовые понятия, используемые в теории ТБД. Проводится критический обзор программных продуктов, реализующих темпоральные технологии, обосновывается выбор платформы для построения темпоральных ИС. Рассматриваются вопросы представления темпоральных объектов в БД, обеспечения их целостности и взаимодействия с ними. Определяются модели и методы, подлежащие использованию в темпоральных ИС на транспорте, проводится их классификация. Формулируются проблемы реализации выбранных моделей и методов, проводится детализация задач настоящего исследования.

В Главе 3 рассматриваются принципы построения ТБД в реляционной среде с использованием *модели данных с абстрактным идентификатором* и темпоральной логики. Особое внимание уделяется представлению темпоральных объектов, обладающих периодичностью и связью с особым календарём. Вводится новая *темпоральная форма Historical-Overlapped*, учитывающая особый режим смены версий темпорального объекта. Решаются вопросы взаимодействия с темпоральными данными. Предлагается *метод обеспечения временного окружения для пользователя*. Решается задача определения актуальной версии темпорального объекта. Для аналитических запросов предлагается *метод трансформации интервальных данных*. Решаются вопросы обеспечения безопасности темпоральных данных.

В Главе 4 приводятся примеры практического применения результатов исследования в ИС на Латвийской железной дороге. Показывается схема применения разработанных методов и моделей в *интерактивной информативно-справочной системе расписания пассажирских поездов*. Исследуются процессы управления хранилищами данных. Особое внимание уделяется процессам трансформации темпоральных данных. Рассматривается практическое применение разработанных моделей и методов в *финансово-статистической информационно-аналитической системе пассажирских перевозок*.

В Заключении представляются основные результаты работы.

10. ОПИСАНИЕ ОСНОВНЫХ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ

10.1. Модели и методы для работы с темпоральными данными в ИС железнодорожного транспорта

На Латвийской железной дороге используется множество ИС, разработанных в разное время по различным технологиям. Все эти системы в различной степени используют темпоральные данные. Для одних темпоральных данных есть специальная поддержка, для других она отсутствует. Темпоральная составляющая проявляется либо в процессах ИС, которые явно или неявно взаимодействуют с временным измерением данных, либо в обрабатываемых данных, которые, даже если они не имеют временного измерения, по сути, являются темпоральными.

Схема взаимосвязи информационных систем предприятия ЛЖД с объектами, имеющими темпоральные характеристики, а также принадлежность этих объектов к одному из видов темпоральных данных показаны на рис. 1. На схеме представлены 17 различных систем: APIKS-2, APNIS, APFIS, APOVS и так далее. Эти системы функционально разделены на 4 прикладные области: грузоперевозки, пассажирские перевозки, экономика и финансы, управление инфраструктурой, кроме того, в отдельную группу выделена система нормативно-справочной информации (НСИ). В процессе настоящего исследования были определены 3 разновидности используемых в ИС на железной дороге темпоральных данных: многоверсионные данные, накопленные исторические данные, временные последовательности (см. рис. 1), которые рассмотрим более детально.

Многоверсионные данные (version-based temporal data) отображают различные состояния объекта в виде набора версий. Со временем множество объектов, участвующих в организации жизнедеятельности железнодорожного транспорта, подвергаются изменениям: меняются маршруты следования поездов, курс валют, тарифы на железнодорожные услуги, тарифы на энергоресурсы, деление железнодорожной сети на линии и участки и так далее. Предыдущие состояния этих объектов (версии) важны для различных аналитических расчетов и решения задач прогнозирования. Чтобы манипулировать версиями, действительными в различные моменты времени, каждое состояние объекта ассоциируется с определенным диапазоном времени – временем его актуальности. Такие состояния объектов могут относиться не только к прошлому и настоящему, но и к будущему, например: запланированное изменение тарифов; расписание, назначенное на следующий сезон, и так далее.

Накопленные исторические данные (historical data) – различные операции, документы и другие однородные данные, постоянно появляющиеся в системе и накапливающиеся в ней (квитанции о продажах билетов, накладные, операции с грузом и т.д.). Такие данные характеризуют функционирование основных железнодорожных процессов во времени. Каждый экземпляр этих данных неразрывно связан с конкретной датой или временем. Именно этот фактор времени является основным измерением в аналитических расчетах.

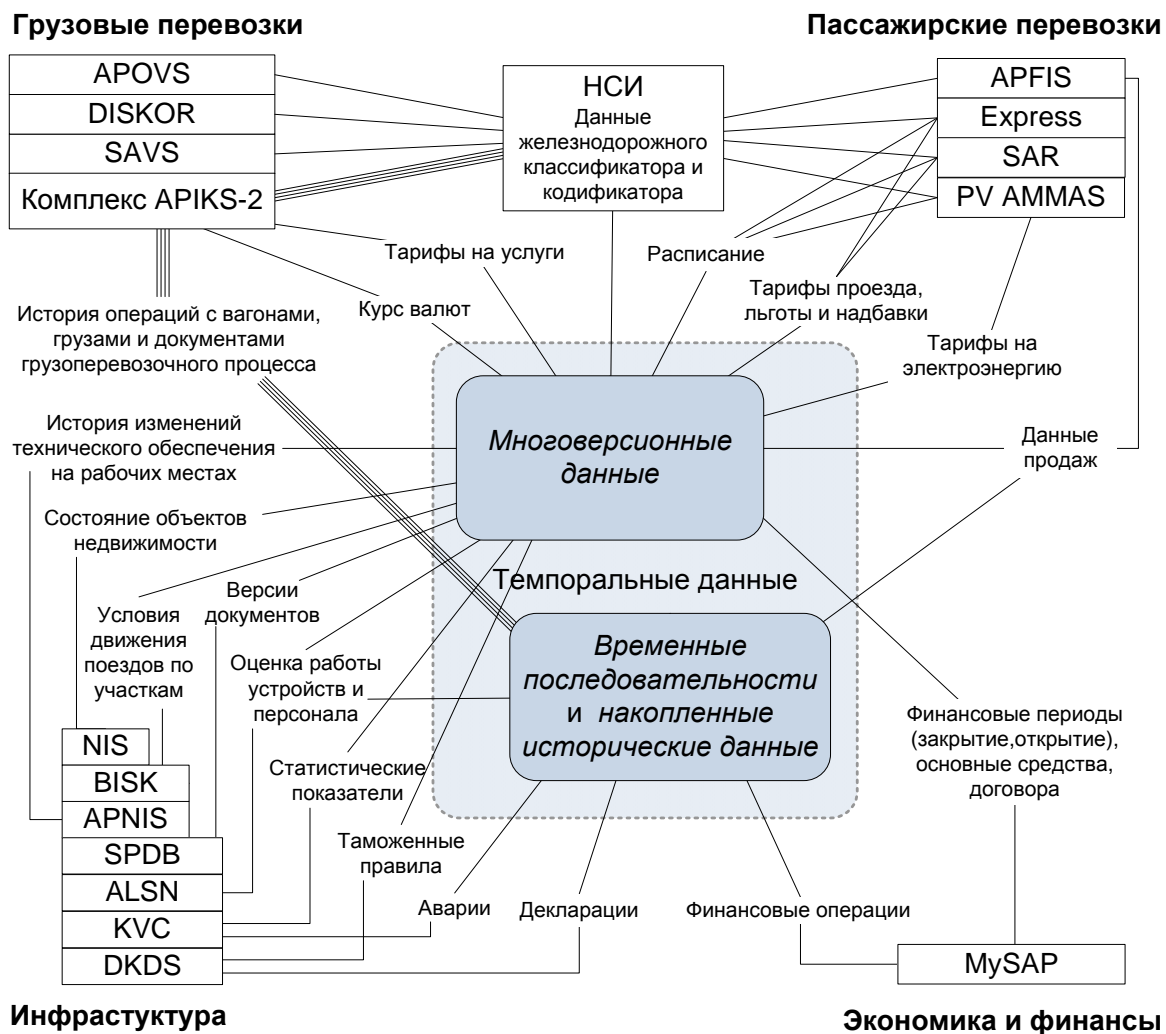


Рис. 1. Темпоральные данные в ИС предприятия ЛЖД

Временные последовательности (time-series based temporal data, streaming data) представляют собой данные, собранные в ходе наблюдений за объектом, производимых последовательно во времени, либо через определенные интервалы, либо непрерывно (например, результаты замеров скорости движения вагонов, координат их местоположения). Наблюдения такого рода можно расположить в хронологическом порядке.

Существуют также и *комбинированные* виды темпоральных данных. Например, в накопленных данных о продажах пассажирских билетов присутствуют не только факты продаж билетов, но и факты их последующих гашений, то есть наблюдается многоверсионность.

Для поддержки темпоральных данных в ИС железнодорожных предприятий необходимы свои модели и методы. Часть из них существовала ранее, некоторые потребовали адаптации, часть была разработана в диссертации. Общая совокупность моделей и методов представлена на рис. 2. Автором было выполнено дополнение и структуризация этого многообразия представленных средств. Анализ ИС, применяемых на железной дороге, показал, что компоненты, участвующие в обслуживании темпоральных данных, располагаются на разных уровнях: системном, логическом и прикладном. Поэтому первым признаком

структуризации является *уровень*. К системному уровню относятся средства СУБД; к логическому уровню отнесены компоненты, позволяющие описать темпоральную модель; к прикладному уровню – модели и методы прикладных систем, учитывающие темпоральные свойства данных. Двойной линией на схеме показаны исследуемые в работе группы моделей и методов. На схеме детально раскрыты только те группы, которые потребовали разработки. Часть методов (что особенно характерно для системного уровня) является общими, некоторые специализированы для темпоральных данных. Белым цветом отмечены известные модели и методы, зеленым – разработанные автором. В целом данная схема отражает план проведённых исследований. При этом основное внимание уделяется логическому уровню, а именно: представлению темпоральных объектов ИС в базе данных и методам взаимодействия с ними. Отметим, что схема имеет универсальный характер и может использоваться при проведении аналогичных исследований в других прикладных областях с некоторыми уточнениями и дополнениями.

Рассмотрим более детально каждый из представленных на схеме уровней.

Системный уровень. Практически все современные промышленные системы не разрабатываются с нуля, как это было свойственно системам 60-70 годов, а используют готовые платформы управления данными. Такие платформы уже включают в себя все необходимые методы физической организации данных. Однако, знание о низкоуровневых методах доступа к данным в выбранной СУБД необходимо для эффективного управления темпоральными данными. На системном уровне предлагаются к использованию пять групп моделей и методов. *Модели хранения данных, методы доступа к физическим объектам хранения, методы повышения скорости доступа к данным* связаны с размещением и организацией данных таблиц внутри файлов на дисковых системах, использованием различных методик индексирования (в т.ч. кластеризации данных), организацией параллельного доступа к данным. *Методы хранения и обработки больших объемов данных* решают аналогичные задачи, но ориентированы на операции с большим объемом данных. Для темпоральных систем последнее обстоятельство является особенно существенным, поэтому эта группа вынесена в отдельный блок. *Механизм транзакций* играет немаловажную роль в реализации целостной части темпоральной модели. Вопросы использования моделей и методов системного уровня рассматриваются во 2-й главе диссертации, Практическое применение в ИС на транспорте представлено в 4-й главе (п. 4.2).

Уровень логического представления. Выбор *темпоральной модели данных* (то есть темпоральной формы) является ключевым моментом на уровне логического проектирования базы данных. Темпоральная модель рассматривается по аналогии с реляционной, которая состоит из трех частей: *структурной, манипуляционной и целостной*. Различия темпоральных моделей в структурной части выражаются в наборе темпоральных атрибутов, целостной – в ограничениях темпоральной целостности, манипуляционной – в алгебре запросов к темпоральным данным. Таким образом, представленные на схеме методы имеют своё специфическое исполнение для каждой темпоральной модели данных.

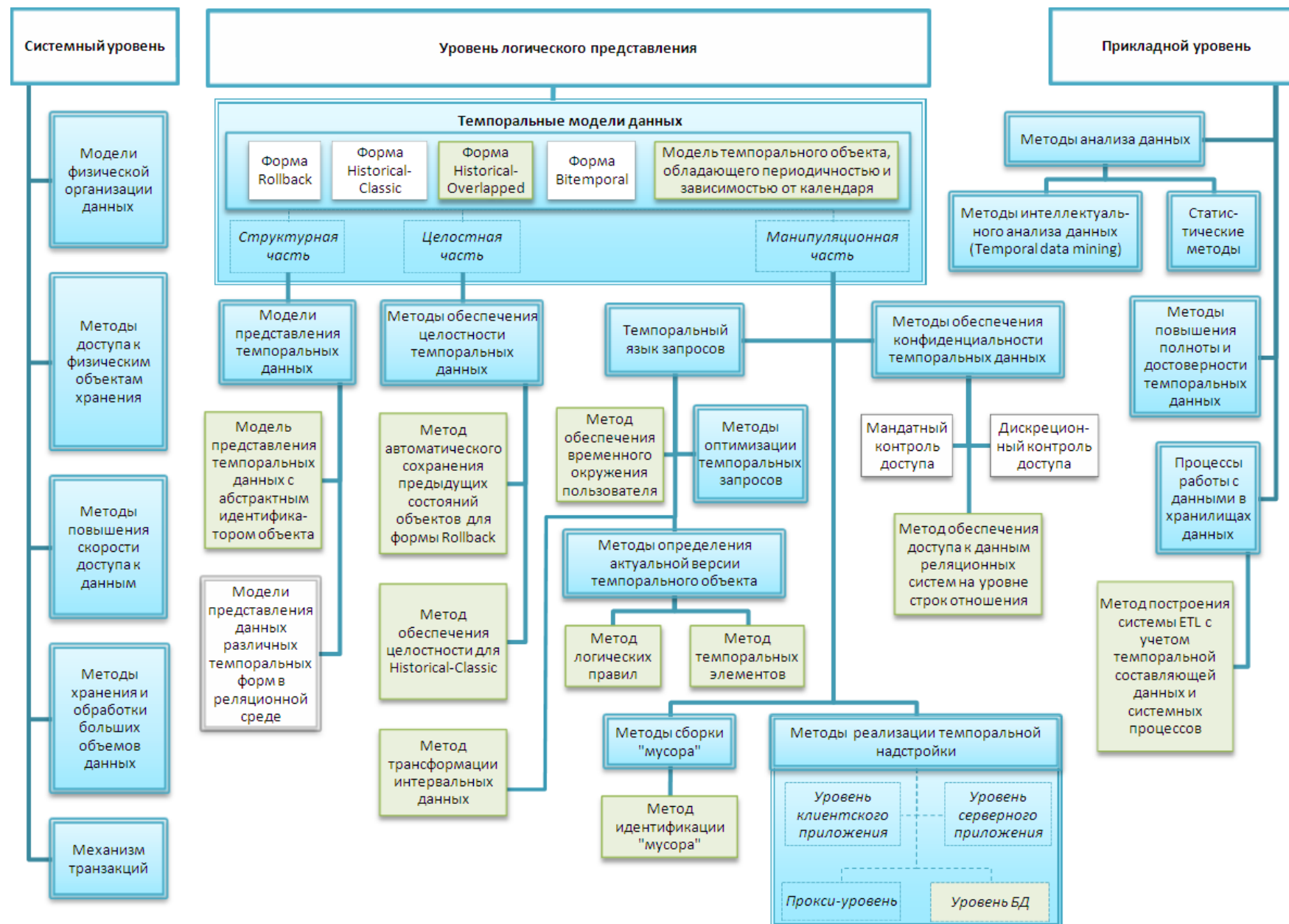


Рис. 2. Совокупность моделей и методов, подлежащих к использованию в темпоральных ИС на жд. транспорте

В *структурной части* в качестве универсальной модели представления темпоральных данных в реляционной среде предложена *модель темпоральных данных с абстрактным идентификатором объекта*, которая позволяет представить темпоральный объект в виде набора реляционных отношений [3, 5]. Реляционные отношения, описывающие темпоральные свойства объекта, могут быть различных форм: *Rollback, Historical-Classic, Bitemporal, Historical-Overlapped*. Первые три формы являются общеизвестными, четвертая форма *Historical-Overlapped* разработана автором и применена к ИС ЛЖД [11]. Разработанная форма учитывает особый режим смены версий темпорального объекта, свойственный некоторым процессам на железнодорожном транспорте. Также на транспорте существует множество объектов, свойства которых обладают периодичностью и связаны с особым (изменённым) календарём. Для описания таких объектов в работе предложена *модель темпорального объекта, обладающего периодичностью и зависимостью от календаря* [11, 12, 13, 15].

Манипуляционная часть включает в себя методы, описывающие корректное взаимодействие с темпоральными данными, а также темпоральную надстройку, обеспечивающую это взаимодействие с учётом целостной части.

Так как специализированные темпоральные СУБД по ряду причин не нашли своего применения в промышленной среде [3, 10], всю логику манипулирования с темпоральными объектами разработчикам транспортных систем приходится создавать собственными силами. Реляционные СУБД не предоставляют или не в полной мере предоставляют средств управления темпоральными данными. В связи с этим манипуляционная и целостная часть темпоральной модели, представленная в реляционной среде, требует расширения соответствующего реляционного аппарата временной логикой. Такую логику можно реализовать на разных уровнях архитектуры системы: на *уровне БД*, независимого промежуточного *прокси-уровня*, *серверного* или *клиентского (пользовательского) приложения*. Таким образом, существует несколько *методов реализации темпоральной надстройки*. Вопросы выбора уровня реализации темпоральной надстройки представлены в работе [10]. В дальнейшем ввиду предпочтительности уровня БД автор рассматривает только этот уровень и предлагает методы темпоральной надстройки, учитывающие особенности поддержки темпоральных данных в реляционной среде в структурном, целостностном и манипуляционном плане. *Метод реализации темпоральной надстройки* в работе представлен *принципами построения темпоральных баз данных в реляционной среде* [3], что является комплексным подходом к построению темпоральных баз данных в среде реляционных СУБД, с которыми можно взаимодействовать не только из темпоральных приложений. Темпоральную надстройку над нетемпоральной БД образует совокупность реализаций представленных на схеме методов логического уровня.

Использование *языка темпоральных запросов* (специальных языковых конструкций запросов к темпоральным данным), *методов оптимизации темпоральных запросов* и особых приёмов обращения к темпоральным данным позволяет избежать тяжеловесных SQL-предложений и повысить эффективность выполнения запросов. Решением многих вопросов в этой

области стал разработанный в диссертационной работе *метод обеспечения временного окружения для пользователя*, который позволяет помещать пользователя в определенную точку во времени и выполнять нетемпоральные запросы к темпоральным данным относительно этого времени. Метод также является средством структурной оптимизации темпоральных запросов.

Каждая темпоральная форма требует своего решения задачи определения актуальной на заданный момент времени версии темпорального объекта. С увеличением количества временных особенностей темпоральной формы возрастает сложность эффективного решения этой задачи. Для её решения в работе предложены два альтернативных *метода определения актуальной версии темпорального объекта: метод логических правил и метод темпоральных элементов* [12, 13, 15].

Большая часть многоверсионных данных в базах данных ИС хранится в интервальном виде. Интервальная форма изначально неудобна для ряда вычислений, но их сложность многократно возрастает, когда присутствуют следующие аспекты: периодичность, особый календарь и необходимость использования в вычислениях данных нескольких взаимосвязанных таблиц. Применение предложенного автором *метода трансформации интервальных данных* решает проблему «неудобного» для вычислений представления данных, превращая интервальный факт во временной ряд.

К *методам обеспечения конфиденциальности темпоральных данных* относятся методы, защищающие темпоральные данные от несанкционированного доступа по операциям чтения и изменения. Вопрос обеспечения конфиденциальности рассматривается в 3-й главе в рамках решения проблемы *управления доступа к темпоральным данным*. Известные подходы к управлению доступом – *дискреционный контроль доступа* и *мандатный контроль доступа* не покрывают всех потребностей разграничительной политики доступа к темпоральным данным. В своём традиционном изложении они не способны учесть темпоральное измерение, которое содержит необходимую для разграничения информацию. Одна из задач данной группы была рассмотрена в работе автора [9], где предложен способ контроля доступа пользователей к данным различных временных периодов с помощью *метода разграничения доступа к данным реляционных систем на уровне строк отношения* [4, 7, 9].

Доступ к темпоральным данным на порядок сложнее и более требователен к используемым вычислительным ресурсам. Повышение эффективности доступа темпоральных данных достигается использованием *методов очищения баз данных от старых ненужных записей (сборка мусора, vacuuming)*. Каждой темпоральной форме соответствует свой *метод идентификации «мусора»* [10].

Целостная часть представлена *методами обеспечения целостности*, к которым относятся методы поддержки темпоральной нормальной формы, целостности значений темпоральных атрибутов и прочих ограничений целостности. Целостная часть также включает поддержку темпоральной функциональности отношений различных форм, например *метод*

автоматического сохранения предыдущих версий объектов для формы *Rollback* [5].

Несмотря на то, что темпоральные отношения являются также и реляционными, известных средств обеспечения целостности реляционных структур недостаточно, так как объекты, содержащиеся в этих системах, многоверсионны, чего в классическом подходе реляционная модель не предусматривает. Здесь необходимы темпорально-ориентированные методы обеспечения целостности. Эта область исследований практически не раскрыта в современной литературе. Вопросы обеспечения целостности и доступности рассматриваются при анализе и разработке темпоральных моделей данных в 3-й главе диссертации.

Прикладной уровень. Методы повышения полноты и достоверности темпоральных данных востребованы как в аналитических, так и в транзакционных системах. Эта категория методов не нуждается в дополнительном комментарии, следует только подчеркнуть, что в алгоритмах методов учитывается темпоральная составляющая данных. В качестве примеров практического применения этих методов назовем: восстановление пропущенных значений, удаление шума в наблюдаемых данных.

Анализ темпоральных данных посредством любых методов анализа, в том числе *статистических методов, методов интеллектуального анализа данных (Data Mining)*, в отличие от обычных методов расширяется умением «понимать» темпоральную структуру данных.

Системы обработки больших объемов данных, такие как хранилища данных, неизбежно связаны с темпоральными данными. Любые модели и методы построения систем обработки больших объемов данных (например, виртуальная модель данных в системах предсказания [6, 8], система подготовки отчетности по пассажирским перевозкам [1, 2]) должны это учитывать. *Процессы работы с данными в хранилищах данных* связаны с вычислением темпоральных агрегатов, работой с временными рядами. *Метод построения системы ETL⁴ с учётом темпоральной составляющей данных и системных процессов*, продемонстрированный в 4-й главе, позволяет упростить реализацию обработки данных и повысить безопасность темпоральных данных.

Далее более детально рассмотрим разработанные автором методы и модели.

10.2. Модель темпоральных данных с абстрактным идентификатором объекта

Учитывая большие вложения в реляционные БД, которые являются основой большинства ИС железных дорог, и накопленный практический опыт автор предлагает использовать реляционную среду для реализации темпоральных БД с разработкой соответствующей темпоральной надстройки и не ставит своей задачей рассмотрение вопроса перехода к специализированным ТСУБД. Решение данной проблемы выполняется по двум направлениям:

⁴ ETL (от англ. *Extract, Transform, Load* – *Выгрузка, Трансформация, Загрузка*) – основные процессы функционирования хранилища данных

разработка моделей темпоральных данных в реляционной среде и разработка временной логики с использованием реляционных средств.

В рамках первого направления автором была разработана *Модель темпоральных данных с абстрактным идентификатором объекта* (Object Abstract Identify, AOID) [3, 5], которая позволяет представить любой темпоральный объект в реляционной базе данных в виде набора реляционных отношений. В этой модели жизненный отрезок объекта описывается через жизненные отрезки всех его свойств, определённых в разных отношениях и имеющих временные атрибуты t_s и t_e , определяющие соответственно время начала и окончания жизненного отрезка. Рассмотрим структурную часть модели – модель представления данных темпорального объекта в виде реляционных отношений.

Предполагается, что объект O темпоральной системы характеризуется уникальным абстрактным идентификатором OID и набором свойств A и может быть представлен в виде:

$$O = (OID, A).$$

Набор свойств объекта $A = \{a_1, a_2, \dots, a_m\}$ подразделяется на два подмножества: множество статических свойств A^s , не подверженных изменениям во времени, и множество динамических свойств A^d , изменяющихся во времени, причём

$$A = A^s \cup A^d, A^s \cap A^d = \emptyset, A^s = \{a_1^s, a_2^s, \dots, a_k^s\}, A^d = \{a_1^d, a_2^d, \dots, a_n^d\},$$

где a_i^s и a_i^d – статическое и динамическое свойство соответственно; k и n – количество статических и динамических свойств соответственно; $m = n + k$ – общее количество свойств объекта O .

Так как значения атрибутов в реляционных базах данных должны быть атомарными, то для представления множественных статических и динамических свойств одного реляционного отношения недостаточно. Представим объект O в виде схемы взаимосвязанных реляционных отношений:

$$O = (R^{OID}, R_1^s, R_2^s, \dots, R_l^s, R_1^d, R_2^d, \dots, R_q^d),$$

где $R^{OID} = R^{OID}(OID, A_0^s)$ – родительское отношение, описывающее абстрактный идентификатор объекта OID и включающее множество A_0^s статических атомарных атрибутов объекта, $A_0^s \subseteq A^s$; OID – ключ отношения R^{OID} ;

$R_1^s, R_2^s, \dots, R_l^s$ – дочерние отношения, описывающие статические свойства объекта A^s (причём $l \leq k$) и имеющие схемы вида:

$R_i^s = R_i^s(OID, A_i^s)$, $i = 1, 2, \dots, l$, в которых A_i^s – подмножество статических свойств объекта, логически объединённых в одном отношении (по принципу оптимизации хранения, необходимости соответствия первой нормальной

форме, информационного и функционального разделения и другим критериям, которые обычно учитывает проектировщик БД), $A_i^s \subseteq A^s$. В случае, если отношение R_i^s описывает атомарное свойство объекта, то OID является ключом, в случае многозначного свойства в состав ключа входят дополнительные атрибуты;

$R_1^d, R_2^d, \dots, R_q^d$ – дочерние отношения, описывающие дискретно изменяющиеся во времени динамические A^d атрибуты (причём $q \leq n$) и имеющие схемы вида:

$$R_i^d = R_i^d(OID, A_i^d, T_i), i = 1, 2, \dots, q,$$

в которых A_i^d – подмножество дискретных динамических атрибутов, логически и по принципу одновременного изменения значений, объединенных в одном отношении, $A_i^d \subseteq A^d$. Посредством такого разделения атрибутов на q отношений достигается темпоральная нормальная форма (TNF); T_i – вектор атрибутов времени, представляющий собой в зависимости от темпоральной формы отношения действительное (VT, Valid Time) или транзакционное измерение (TT, Transaction Time) или оба сразу, $T_i \subseteq T$, где $T = \{t_s^{VT}, t_e^{VT}, t_s^{TT}, t_e^{TT}\}$ – множество атрибутов временных измерений. В случае если отношение R_i^d описывает атомарные свойства объекта, то, в зависимости от временной формы, ключ отношения представляет собой пару (OID, t_s^{TT}) или (OID, t_s^{VT}) . В случае множественных свойств в состав ключа входят дополнительные атрибуты.

Общий вид модели темпоральных данных с абстрактным идентификатором объекта, представленной в виде ER схемы, показан на рис. 3.

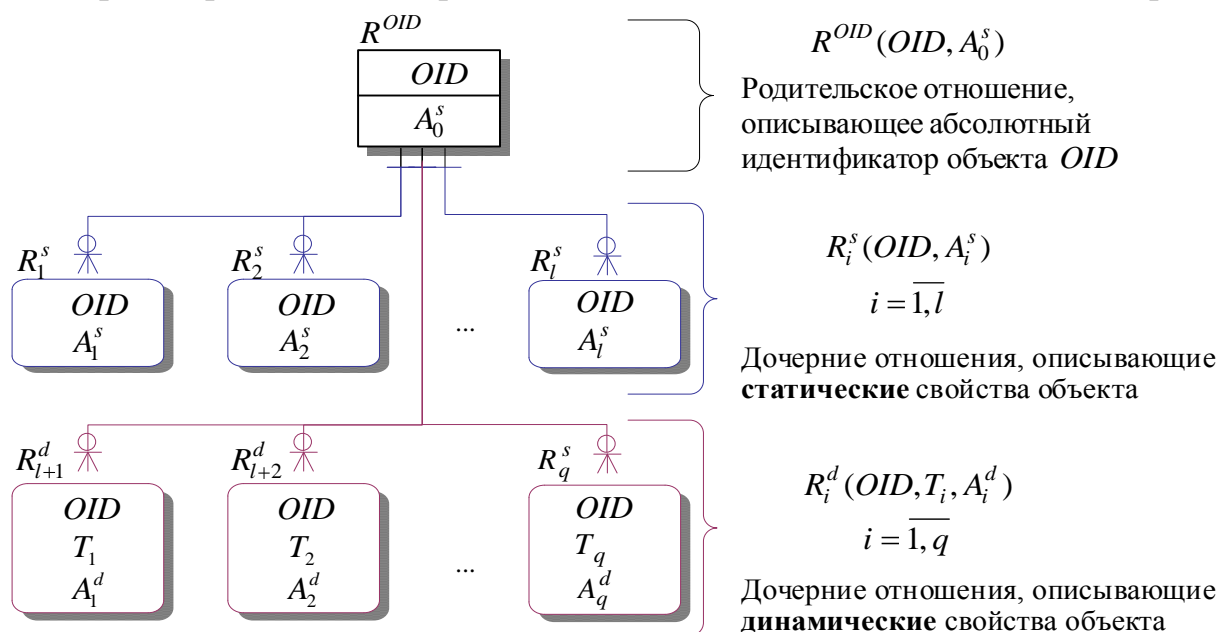


Рис. 3. Модель данных с абстрактным идентификатором объекта в виде ER схемы

10.3. Модель темпорального отношения с пересекающимися жизненными отрезками **Historical-Overlapped**

Одна из существенных проблем, возникающих в рассматриваемых задачах ИС на железнодорожном транспорте, связана с пересечением периодов жизненных циклов объектов. В отличие от классического понятия жизненного цикла и принципа смены версий темпоральных объектов, жизненный цикл версии объекта в таких задачах может быть разбит на несколько отрезков, что означает возможность версии объекта несколько раз за время наблюдения входить и выходить из «теневого зоны». Примером системы с такими особенностями является система учёта расписания пассажирских поездов, рассмотренная в работах автора [5, 12, 13, 15]. Темпоральным объектом в ней является расписание поезда. Одна и та же его версия может быть актуальной на нескольких временных отрезках. Например, основной график движения поездов может на несколько дней подменяться другим графиком, адаптированным к проведению ремонтных работ на путях, а после их завершения опять вступать в силу.

Использование известных форм темпоральных отношений Snapshot, Historical, Rollback, Bitemporal (см. рис. 4) для описания темпоральных объектов с такими особенностями неэффективно, что выражается не только в неэкономном использовании дискового пространства, но и в повышенном риске нарушения целостности темпоральных данных при их изменении. В условиях постоянного увеличения объемов данных и повышенных требованиях к целостности и доступности данных в транспортных системах перечисленные недостатки становятся критичными [11].

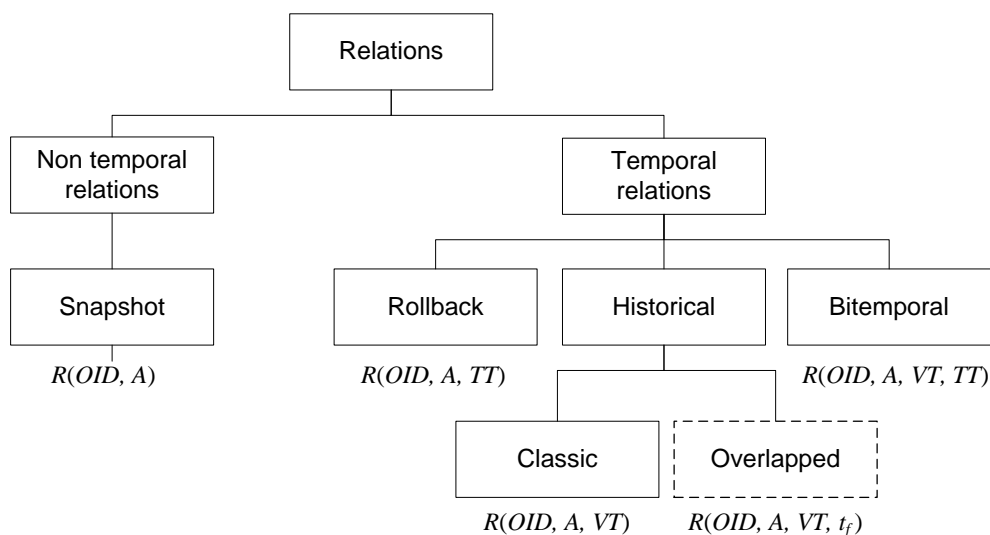


Рис. 4. Форма Historical-Overlapped в общей классификации форм отношений темпоральных БД

Автором предложена особая темпоральная форма данных, не обладающая указанными выше недостатками, – форма отношения с пересекающимися временными отрезками Historical-Overlapped (на рис. 4 показана пунктиром), рассмотренная в работах автора [11, 12, 13, 15]. Она включает в себя основные

свойства известных типов темпоральных отношений, а именно возможность существования нескольких версий объекта, которые последовательно становятся актуальными. Вместе с этим исследуемая форма позволяет версиям объекта становиться актуальными более одного раза.

Схема темпорального отношения с пересекающимися временными отрезками Historical-Overlapped представлена следующим образом:

$$R(OID, A, VT, t_f),$$

где VT – диапазон действия i -й версии, характеризуемый датами начала и завершения: $VT = [t_s^{VT}, t_e^{VT}]$, t_f – время фиксации транзакции (соответствует рассмотренному выше атрибуту t_s^{TT}). Время фиксации транзакции образуется в момент попадания или изменения записи в базе данных. На основе значений этого атрибута вычисляется актуальная версия объекта OID .

Существование двух и более версий с пересекающимися временными диапазонами не противоречит целостности отношения Historical-Overlapped. То есть в отношении R могут существовать два кортежа r и p , таких что:

$$(r.OID = p.OID) \wedge (r.VT \cap p.VT \neq \emptyset).$$

В качестве примера отношения с пересекающимися временными отрезками рассмотрим задачу хранения расписания отправления поездов со станции.

Пример 1. Задача хранения расписания отправления поездов со станции.

Расписание движения большинства поездов имеет продолжительный характер, однако в результате различных обстоятельств возможны плановые и внеплановые изменения времени отправления. Задача состоит в том, чтобы в любой момент времени для любого дня в прошлом и будущем определить действующее в этот день время отправления заданного поезда с наблюдаемой станции.

На рис. 5 показан пример сценария изменения расписания поезда. Рассмотрим его более подробно. Для поезда с номером 123Р на летний период с 01.05 по 30.08 было назначено базовое сезонное расписание (*Base*). Впоследствии, в связи с необходимостью проведения ремонтных работ в период с 03.07 по 15.07, расписание поезда было изменено (*Repair works*). А на день города 06.07 было назначено особое расписание (*City anniversary*).

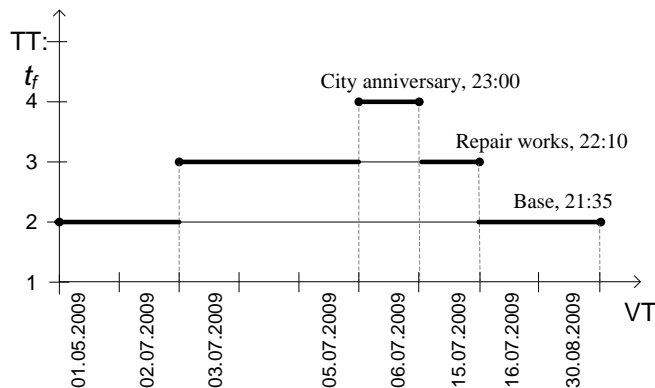


Рис. 5. Пример пересечения жизненных отрезков расписания

На рис. 6 показано, как информация из рассматриваемого примера может быть представлена в двух различных формах: в классическом отношении с непересекающимися временными отрезками Historical-Classic и в отношении с пересекающимися временными отрезками Historical-Overlapped. Атрибуты действительного времени t_s^{VT} и t_e^{VT} определяют соответственно начало и окончание периода действия расписания; атрибут t_f – время транзакции.

Historical-Classic				Historical-Overlapped				
Train OID	Departure time	t_s^{VT}	t_e^{VT}	Train OID	Departure time	t_s^{VT}	t_e^{VT}	t_f
123P	21:35	01.05.2009	02.07.2009	123P	21:35	01.05.2009	30.08.2009	1
123P	22:10	03.07.2009	05.07.2009	123P	22:10	03.07.2009	15.07.2009	2
123P	23:00	06.07.2009	06.07.2009	123P	23:00	06.07.2009	06.07.2009	3
123P	22:10	07.07.2009	15.07.2009					
123P	21:35	16.07.2009	30.08.2009					

Рис. 6. Отношение «Расписание отправления поезда со станции» в формах Historical-Classic и Historical-Overlapped

Данный пример показывает, что отношение с пересекающимися временными отрезками более экономно с точки зрения количества кортежей, а также без введения дополнительных атрибутов более ясно выражает семантику задачи. Выигрыш в передаче семантики достигается тем, что в отношении Historical-Overlapped кортеж представляет собой еще и факт появления замены, а не просто очередность смен расписаний, как это делается в отношении Historical-Classic.

Принцип управления версиями объектов в отношениях Historical-Overlapped отличается от управления в Historical-Classic. Отличительной особенностью Historical-Overlapped является то, что появлению новой версии объекта или изменению существующей не предшествует обработка значений временных атрибутов соседних (предыдущей и/или последующей) версий объекта. Напротив: в отношении Historical-Classic создание новой версии объекта или модификация существующей связаны с обработкой границ жизненных отрезков других версий, жизненный отрезок которых оказался частично или полностью перекрыт в результате модификации (в большинстве случаев это относится к предыдущей и следующей версии объекта).

Задача идентификации актуальной на момент t версии объекта в форме Historical-Overlapped связана с более сложным доступом к данным, так как помимо проверки вхождения t в диапазон действия версии объекта $[t_s^{VT}, t_e^{VT}]$ необходимо выбрать наиболее «свежую» версию объекта, используя атрибут t_f .

Обобщенные результаты сравнения форм отношений Historical-Classic и Historical-Overlapped представлены в табл.1. При сравнении учитывались две возможные ситуации реального моделирования: когда одна и та же версия

объекта не становится актуальной более одного раза и когда одна и та же версия объекта актуальна на нескольких временных отрезках.

Таблица 1. Результаты сравнения форм отношений *Historical-Classic* и *Historical-Overlapped*

Ситуация \ Показатель	Одна и та же версия объекта не становится актуальной более одного раза: $g = h$		Одна и та же версия объекта актуальна на нескольких временных отрезках: $g < h$	
	Historical-Classic	Historical-Overlapped	Historical-Classic	Historical-Overlapped
Размер темпорального отношения (количество кортежей)	g	g	$[g + 1; 2g - 1]$	g
Целостность при изменении данных (количество операций с кортежами при появлении новой версии объекта)	0	0	$[0; u + d + i]$	0
Доступность при идентификации актуальной на момент t версии объекта (количество обрабатываемых кортежей)	$\{0, 1\}$	$\{0, 1\}$	$\{0, 1\}$	$[0; g(t)]$

Поясним обозначения, используемые в табл. 1:

g – количество версий объекта;

h – количество периодов актуальности версии, $g \leq h$;

$u \in \{0, 1, 2\}$ – количество кортежей, отвечающих частично перекрытым жизненным отрезкам; эти u кортежей должны быть изменены;

$d \in \{0, 1, \dots, h\}$ – количество кортежей, отвечающих полностью перекрытым жизненным отрезкам; эти d кортежей должны быть удалены или помечены, как недействующие;

$i \in \{0, 1\}$ – количество кортежей, отвечающих жизненным отрезкам, которые полностью перекрывают жизненный отрезок новой версии; эти i кортежей должны быть добавлены, причём $(i = 1) \Rightarrow (d = 0)$; $(d > 0) \Rightarrow (i = 0)$;

$g(t)$ – количество версий объекта, действительных в момент t .

10.4. Реализация временной логики в рамках реляционной модели

Вторым направлением задачи построения темпоральной надстройки является разработка временной логики в рамках реляционной модели. Для ее решения автором было предложено расширение манипуляционной и целостной части реляционной модели. Основными задачами этого расширения являются:

- поддержка хронологической многоверсионности и обеспечение темпоральной целостности;

- выполнение операций удаления DELETE, обновления UPDATE и расширенной операции выборки SELECT с учётом временных свойств объекта;
- поддержка темпоральной восходящей совместимости с действующими нетемпоральными приложениями;
- обеспечение доступа ко всем предыдущим и последующим версиям объекта;
- гарантирование существования только одной актуальной версии объекта в любой момент времени.

Остановимся на основных особенностях реализации темпоральной надстройки.

Для сохранения логики действующих приложений необходимо разрешить операции удаления DELETE и изменения UPDATE объекта (т.е. его свойств), причём при выполнении этих операций ни одно состояние объекта не должно теряться. В этом случае предлагается следующий механизм выполнения операций модификации объекта. В момент времени t_o изменения i -го состояния объекта компонент темпоральной надстройки переводит неактуальную версию объекта в «теневую» зону с автоматической фиксацией времени окончания периода жизненного цикла $t_e^{(i)} = t_o$. При этом для операции DELETE процесс завершается, а в случае операции UPDATE темпоральная надстройка создает новый кортеж отношения (новую запись), то есть актуальную $(i+1)$ -ю версию объекта с временем начала жизненного цикла $t_s^{(i+1)} = t_o$ (см.рис. 7). Такая временная логика соответствует темпоральной форме Rollback.

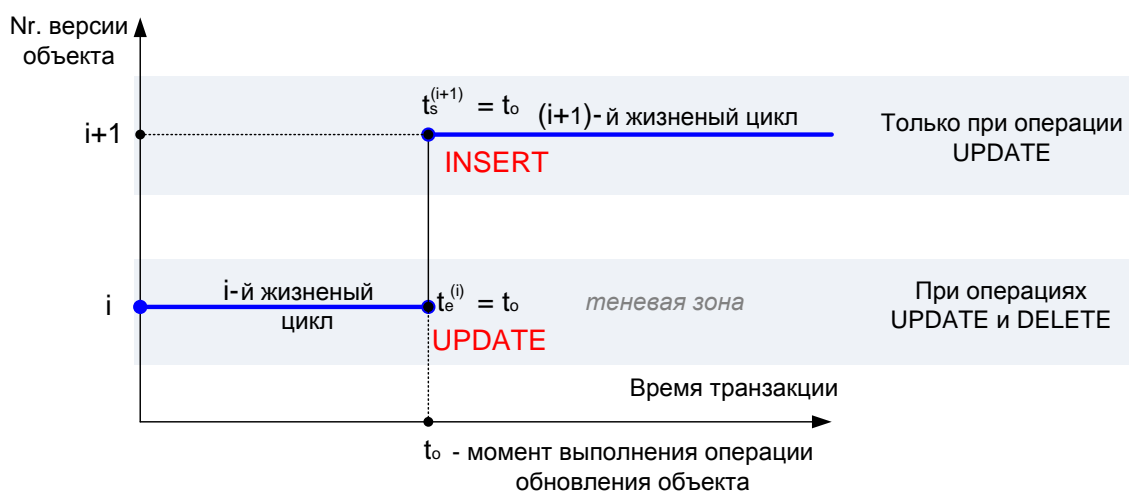


Рис. 7. Диаграмма изменения состояния объекта в темпоральной форме Rollback

Предложенная автором схема взаимодействия с базой данных, реализующая вышеописанную логику при выполнении операции UPDATE, показана на рис. 8. Для её реализации использованы стандартные средства активных баз данных: язык SQL, представления данных (View) и триггеры.

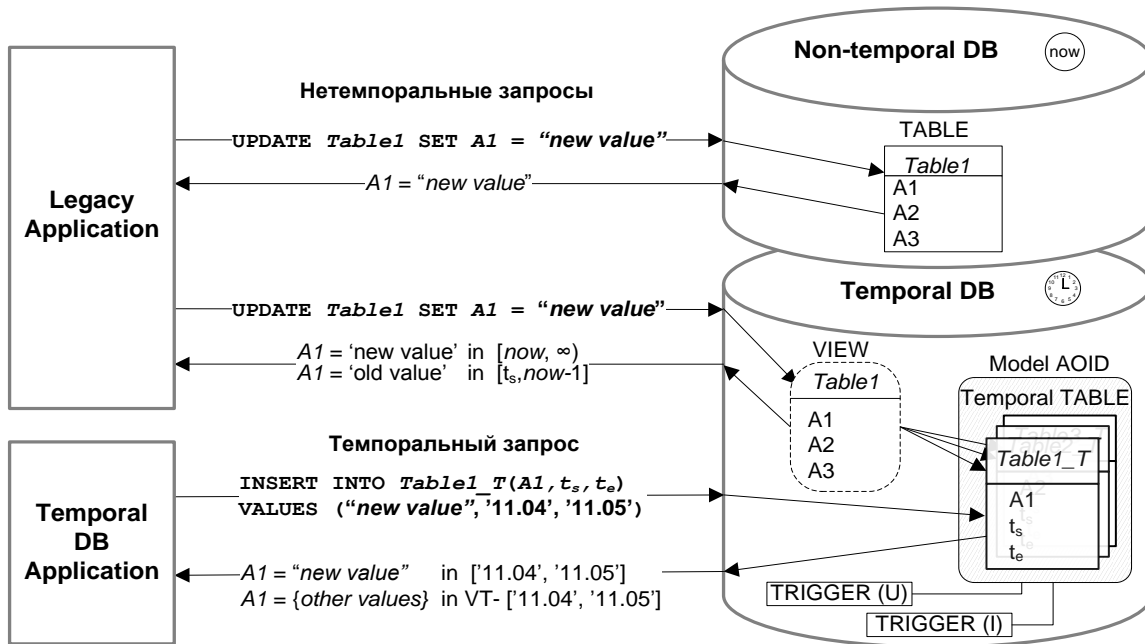


Рис. 8. Поддержка запросов обновления временных данных

При наделении объектов ИС временным измерением в БД происходят серьёзные структурные изменения, начиная от состава первичного ключа и вплоть до декомпозиции отношений. Такие изменения не могут не влиять на логику функционирования приложений. Поддержка существующей логики для действующих приложений обеспечивается сохранением интерфейса их взаимодействия с БД.

Для решения задачи сохранения «старого» интерфейса в БД, расширенной временными свойствами, предложено сформировать представления данных, имеющие структуру и идентификаторы элементов исходной БД до структурных изменений. Тогда всем элементам БД, подвергшимся структурным изменениям, необходимо присвоить новые имена. В примере, представленном на рис. 8, при временном расширении таблица *Table1* была преобразована в набор таблиц, спроектированных согласно модели с AOID: *Table1_T0*, *Table1_T1*... и т.д. в зависимости от количества динамических атрибутов. А с её названием *Table1* создано представление, отображающее хранимые данные в их первоначальном виде. Представление *Table1* осуществляет композицию данных модели OID и срез всех временных таблиц, содержащий актуальные значения данных на момент запроса. Приложение (*Legacy Application*) взаимодействует с *Table1* посредством запросов, используемых до временного расширения таблицы, и получает результаты, соответствующие её первоначальной структуре. Генерируемый приложением запрос на модификацию (UPDATE) не несет никаких данных о временном диапазоне для нового значения. Временная надстройка автоматически перехватывает этот запрос и посредством триггера *Trigger(U)* дополняет его серией действий по сохранению предыдущей версии и установкой значений временных атрибутов для новой.

На рис. 8 также отражена технология обработки запросов, посылаемых приложением, которое учитывает временный характер данных (*Temporal DB Application*). Модификация данных в этом случае происходит путём вставки

новой версии, для которой указаны все необходимые временные метки. Темпоральная надстройка здесь выполняет другую роль: посредством триггера *Trigger(I)* она осуществляет поддержку темпоральной целостности данных, производя обновление временных диапазонов смежных версий при возникновении ситуаций их наложения, и, при необходимости, разбивая диапазон предыдущей версии на два отрезка.

Таким образом, предлагаемый метод позволяет наделить базу данных сразу двумя различными интерфейсами: классическим реляционным (нетемпоральным) и темпоральным. Первый обеспечивает темпоральную восходящую совместимость с уже действующими нетемпоральными приложениями, предоставляя данные, актуальные на момент взаимодействия, второй представляет темпоральные данные в исходном виде. При этом через нетемпоральный интерфейс поддерживаются все операции модификации, а невидимая для приложения темпоральная надстройка автоматически поддерживает темпоральную целостность данных. Данный метод рассмотрен и применён на практике в 3-й главе диссертации и в работах [3, 5].

Тем не менее для ряда действующих транспортных ИС взаимодействия лишь с актуальными на момент запроса значениями недостаточно. В первую очередь это относится к системам обработки накапливаемых исторических данных. Каждый экземпляр таких данных (платеж, договор, транспортная операция и т.п.) связан с определенным моментом времени и, следовательно, с актуальной для него нормативно-справочной информацией. Однако посылаемые действующими приложениями запросы к базе данных НСИ не учитывают темпорального измерения данных, а значит, не способны настроиться на необходимый временной срез. Решением проблемы перемещения нетемпоральных запросов в определенную точку во времени является использование *метода обеспечения временного окружения для пользователя*. Основой метода является смещение точки текущего времени *now* и помещение пользователя (или приложения) в определенный момент во времени. Выполнение всех его запросов к данным происходит относительно смещённой точки. Этот момент времени задаётся в специально определяемом временном окружении пользователя и называется *переменной окружения*. Таким образом, метод позволяет выполнять нетемпоральные запросы к многоверсионным данным в различных временных срезах. Покажем основной принцип его работы на следующем примере.

Допустим, существует форма нетемпорального запроса $Q(O)$ к БД, возвращающего данные объекта O . После наделения БД темпоральным измерением в ней теперь хранится не одна версия состояния объекта O , а три: v_1 , v_2 и v_3 (как это показано на рис. 9). Рассмотрим два варианта работы запроса: при неопределенной переменной окружения пользователя и когда она равна t_1 – моменту времени из прошлого. В первом случае темпоральная надстройка выполнит запрос относительно текущего времени (*now*) и результатом запроса $Q(O)$ будет являться версия объекта v_2 , актуальная на момент выполнения запроса. Во втором случае выполнится запрос относительно установленной

временной точки во временном окружении, равной t_1 и в результате – версия v_1 , актуальная на момент t_1 .

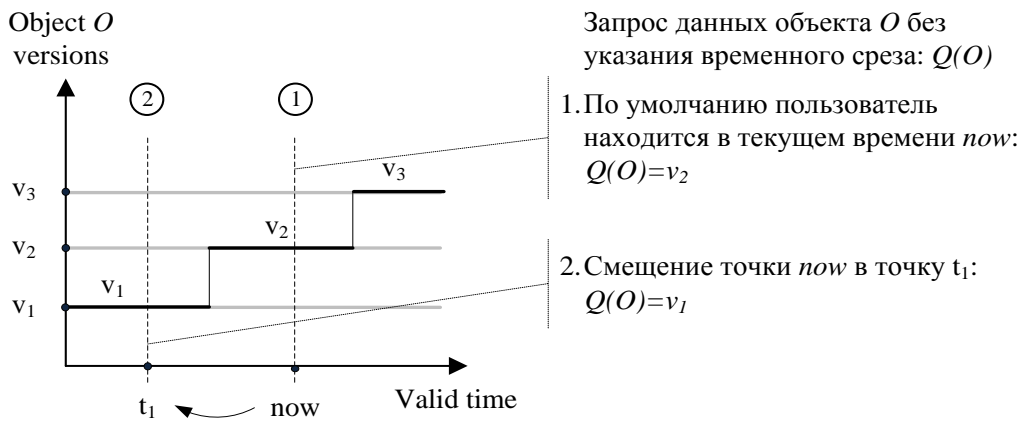


Рис. 9. Смещение точки now для запросов пользователя

На рис. 10 показан переход от нетемпоральной базы данных к темпоральной для действующего приложения. На схеме видно, что приложение генерирует одни и те же нетемпоральные запросы, однако, помимо сохранения нетемпорального интерфейса, новая темпоральная БД способна обрабатывать запросы в различных временных срезах, что необходимо для более точного функционирования приложения. В рассматриваемом примере обрабатываются запросы двух различных пользователей, которым определены различные переменные окружения. Видно, что на один и тот же запрос пользователи в ответ получают разные значения, каждое из которых актуально на момент времени, заданный в переменной окружения пользователя.

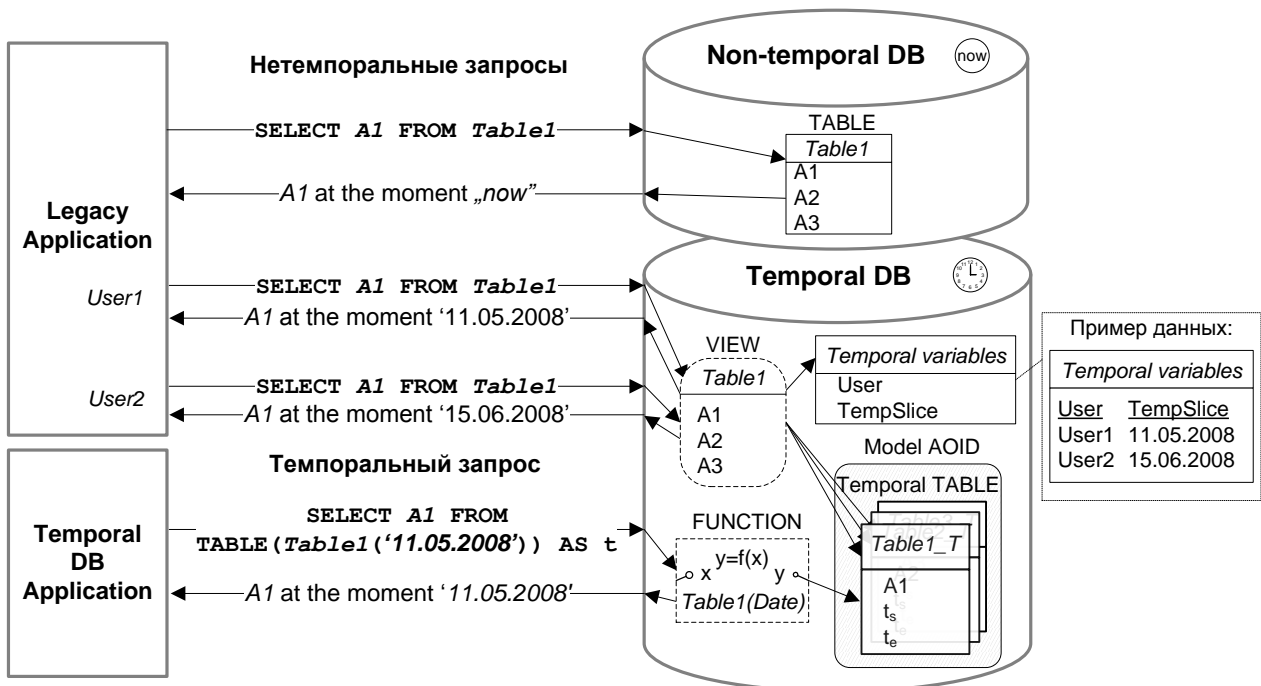


Рис. 10. Поддержка интерфейса нетемпоральной операции чтения данных с использованием переменной окружения пользователя

С помощью данного метода можно оперативно манипулировать временным срезом, в котором необходимо работать пользователю, не меняя код приложения и не используя в запросах дополнительных условий. При изменении переменной окружения запросы автоматически и мгновенно начинают работать в другом временном слое. Отметим, что при использовании данного метода все пользователи могут работать независимо, так как для каждого определено своё временное окружение.

Применение предложенного метода целесообразно и при разработке новых ИС. Его использование позволяет значительно упростить код запросов к многоверсионным данным, так как избавляет от необходимости задавать дополнительные условия для попадания в определенный временной слой. Это снижает вероятность появления ошибок в коде запросов, а, следовательно, повышает достоверность получаемых результатов.

Временное окружение пользователя предложено использовать также при вычислении темпоральных агрегатов, которые являются непременной составляющей любой аналитической обработки. Выполнение операций темпоральных агрегаций связано с определением временной области (периода с конкретными датами начала и окончания), для которой вычисляются агрегаты. Основой метода является указание временного диапазона, в рамках которого действует пользователь, и его дальнейшее принудительное использование в вычислениях. В сложных расчетах, таких как подготовка отчетности по железнодорожным пассажирским перевозкам [1, 2, 6], где используются серии вычислений взаимосвязанных агрегатов, исключение из запроса опции спецификации временного диапазона не только позволит упростить код запросов, но и значительно снизит вероятность появления ошибок.

Временное окружение пользователя предложено использовать также как средство повышения безопасности темпоральных данных. Особая переменная окружения позволяет ограничить допустимую область действий пользователя множеством данных, относящимся к заданному диапазону времени. Использование этой переменной временного окружения описано в п. 10.7 «Метод контроля доступа к темпоральным данным на уровне кортежей отношения» и в работах [9, 10].

10.5. Модель периодичности в задачах железнодорожного расписания

Главные проблемы хранения расписания движения поездов в базах данных связаны с наличием множества его версий, которые обусловлены сезонными циклами смены расписаний, днями недели, плановыми и неплановыми ремонтными работами, переносами праздничных и рабочих дней и другими факторами. Основная проблема заключается в сложности обеспечения достоверности данных и оперативности внесения изменений в расписание. В условиях, когда изменения происходят часто, эти задачи становятся крайне актуальными.

Для решения вышеназванных проблем в рамках данного исследования разработана модель темпорального объекта, обладающего периодичностью и

особым календарём и предложены методы определения актуальной версии объекта: метод логических правил и метод темпоральных элементов (см. п. 10.6 и работы [12, 13, 15]). Полученные результаты применены при создании ИС расписания движения поездов внутреннего сообщения на Латвийской железной дороге.

10.5.1. Использование принципа периодичности в расписании поездов

На практике поезд может иметь одновременно несколько действующих расписаний в течение длительного промежутка времени (см. главу 1 диссертации). При этом каждое расписание поезда формируется с учётом дней курсирования с использованием различных признаков периодичности, например: по рабочим дням, по выходным, по определённым дням недели, в первый рабочий день недели, в последний выходной день недели (причём это может быть и понедельник), по чётным или нечётным дням и т.п.

В качестве примера рассмотрим многовариантное расписание поезда, представленное на рис. 11. Как видно из данного рисунка, для некоего поезда имеется два базовых расписания, одно для рабочих дней – wd , другое для выходных дней – rd . Оба расписания являются актуальными во временном интервале $[t_1, t_4]$, но в разные дни недели. Затем для проведения плановых ремонтных работ по вторникам и четвергам (Tue, Thu) вносится изменение в расписание данного поезда в период времени с t_2 по t_3 . Таким образом, во временном интервале $[t_2, t_3]$ мы имеем уже три актуальных версии расписания, действующих в дни, которые соответствуют признакам wd , rd и Tue, Thu , причём в один день актуальным является только одна версия расписания wd , rd или Tue, Thu .

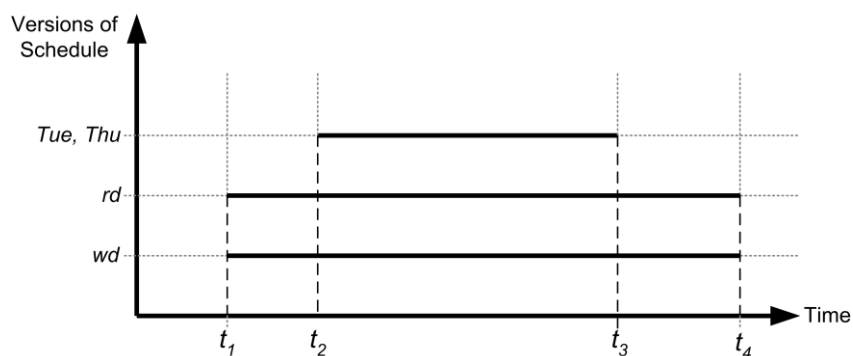


Рис. 11. Множество действующих версий расписания поезда: wd , rd и Tue, Thu

Последнее утверждение проиллюстрируем с помощью диаграммы на рис. 12, где по оси „Day of week” показаны дни недели, соответственно 1 – понедельник, 2 – вторник и т.д. Время в данной системе имеет дискретный характер, уровень гранулярности расписания равен одному дню. Как видно из данного рисунка, в разные дни недели актуальным расписанием поезда является одна версия из трёх, а остальные две в этот момент находятся в «теневой» зоне. Причём расписание поезда по вторникам и четвергам (признак Tue, Thu) перекрывает расписание рабочего дня, признак которого wd .

Однако помимо периодических версий расписания поезда существуют и разовые изменения расписания, связанные с особым (измененным) календарём. К таким изменениям можно отнести праздничные и дополнительно назначенные выходные дни или отменённые выходные дни. Выделим также особый случай переноса расписания, когда в календаре рабочий день меняется местами с выходным днём. Так, например, в 2007 году в Латвии понедельник 30 апреля (рабочий день) был перенесён на субботу 14 апреля (выходной день). В результате поезда 30 апреля курсировали по расписанию выходного дня – 14 апреля, а 14 апреля все поезда имели расписание рабочего дня – 30 апреля.

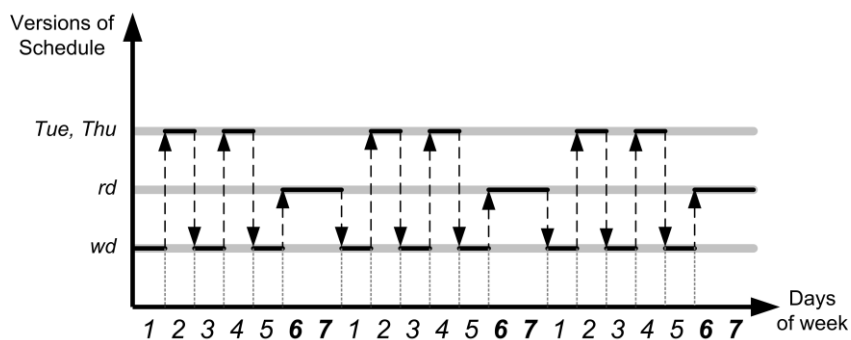


Рис. 12. Периодичность в расписании поезда

Подобные переносы или назначения вызывают аномалии (парадоксы). Например, понедельник 19 ноября 2007 года был объявлен дополнительным выходным днем, и в этот день все поезда отправлялись по расписанию выходного дня (18 ноября), что привело к сбою периодичности, как это показано на рис. 13. Следует добавить, что несколько поездов с периодичностью (воскресенье) были отменены 18 ноября из-за того, что обычное воскресенье – это ещё и последний выходной день, а дополнительные воскресные поезда рассчитаны на «вывоз» пассажиров из традиционных мест отдыха в выходные дни.

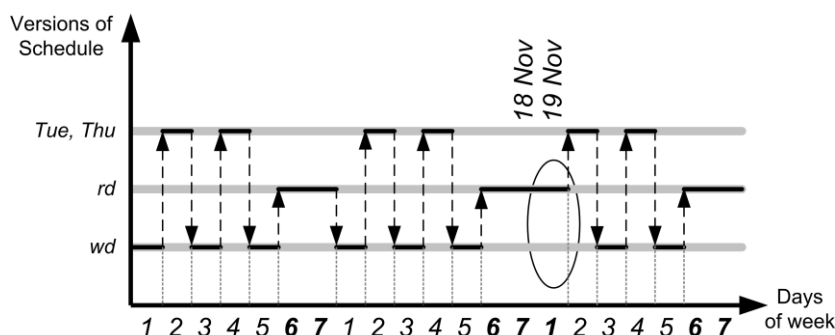


Рис. 13. Сбой периодичности в расписании поезда

Для корректной работы системы расписания, в которой должны учитываться рассмотренные выше нетривиальные правила, требуется чёткая формализация постановки задачи и описания темпоральной модели расписания.

10.5.2. Математическое описание системы железнодорожного расписания

Для описания объектов железнодорожного расписания введём два базовых множества:

$S = \{s_1, s_2, \dots, s_k\}$ – множество всех станций железной дороги, где k – мощность множества S ;

$N = \{n_1, n_2, \dots, n_m\}$ – множество поездов железной дороги, где m – мощность множества N .

Расписание поезда с номером $n \in N$ будем определять вектором вида:

$$v = \{\langle s_1^{(n)}, T_1 \rangle, \langle s_2^{(n)}, T_2 \rangle, \dots, \langle s_\alpha^{(n)}, T_\alpha \rangle\},$$

где пара $\langle s_j^{(n)}, T_j \rangle$ определяет j -ю остановку поезда, а именно станцию $s_j^{(n)} \in S$ и время отправления поезда T_j ; α – количество остановок поезда, идущего по расписанию v .

Для поезда с номером n может храниться несколько версий расписания, характеризующихся периодом действия, временем назначения (фиксации) и периодичностью. Соответственно введём следующие параметры расписания:

t_s и t_e – время начала и окончания периода действия данной версии расписания соответственно;

t_f – время фиксации версии расписания в БД;

C – признак периодичности, используемый для определения дней, в которые будет действовать данное расписание.

Признак C представляет собой логическое выражение, состоящее из одного или нескольких элементарных признаков периодичности, которые соединены знаками логических операций \vee, \wedge и \neg . Каждый элементарный признак периодичности p_i соотносится с предикатом $p_i(x)$, определяющим принадлежность дня к конкретной группе, например, «день является четным днем месяца», которая может быть *true* или *false*. Элементарные признаки периодичности образуют такой набор признаков, на базе которого можно строить логические выражения, задающие любую более сложную периодичность:

$$C(x) = f(p_i(x), i = \overline{1, \lambda}), \quad (1)$$

где $p_i(x) \in B \equiv \{true, false\}$; λ – арифметическая сложность высказывания C .

Нами будут рассматриваться различные варианты признака периодичности, задаваемые множеством $P = \{p_1, p_2, \dots, p_\pi\}$, где π – мощность множества; элементами множества являются следующие высказывания: *ed* – «любой день недели» (используется, если поезд курсирует ежедневно); *wd* – «является рабочим днем» (поезд курсирует по рабочим дням); *rd* – «является выходным днем»; *Mon, Tue, Wed, Thu, Fri, Sat, Sun* – «является соответствующим днем недели»; *pd* – «является четным днем месяца»; *nd* –

«является нечётным днем»; lh – «последний выходной день недели»; fw – «первый рабочий день недели» и др.

Предикат $p_i(x)$ определяет для дня x свойство «являться» днём с признаком p_i . Множество истинности предиката $p_i(x)$ на области определения $x \in [t_s, t_e]$ представляет собой множество всех дней с признаком p_i в диапазоне $[t_s, t_e]$. Например, определение множества истинности предиката $Mon(x)$ для $x \in [01.05.2007, 31.05.2007]$ выглядит следующим образом:

$$\{x | x \in [01.05.2007, 31.05.2007], Mon(x)\} = \\ \{07.05.2007, 14.05.2007, 21.05.2007, 28.05.2007\}.$$

Таким же образом множество истинности предиката $C(x)$ для заданной версии представляет собой множество дней в диапазоне $[t_s, t_e]$, в которые эта версия является актуальной.

Для идентификации конкретной версии расписания для поезда с номером $n \in N$ будем использовать кортеж вида $\langle n, C, t_f, t_s, t_e \rangle$.

Тогда i -я версия расписания $v_i^{(n)}$ для поезда с номером $n \in N$ будет определяться кортежем вида:

$$v_i^{(n)} = (\langle n, C, t_f, t_s, t_e \rangle, \langle s_1^{(n)}, T_1 \rangle, \langle s_2^{(n)}, T_2 \rangle, \dots, \langle s_\alpha^{(n)}, T_\alpha \rangle). \quad (2)$$

Пусть $V^{(n)} = \{v_1^{(n)}, v_2^{(n)}, \dots, v_g^{(n)}\}$ – множество версий расписания поезда с номером $n \in N$, где g – мощность множества, равная количеству версий расписания поезда n , тогда $V = \{V^{(n_1)}, V^{(n_2)}, \dots, V^{(n_m)}\}$ – множество версий расписаний всех поездов.

Далее введём множества, характеризующие исключения в периодичности, назначенные директивно:

$EX(C) = \{ex_1^{(C)}, ex_2^{(C)}, \dots, ex_\theta^{(C)}\}$ – множество дней-исключений $ex_j^{(C)}$, которым присваивается признак C , где θ – мощность множества $EX(C)$. Характерными примерами множества $EX(C)$ служит множество $EX(wd) = \{ex_1^{(wd)}, ex_2^{(wd)}, \dots, ex_w^{(wd)}\}$, задающее дополнительно назначенные рабочие дни признаком периодичности $C = wd$, и множество $EX(rd) = \{ex_1^{(rd)}, ex_2^{(rd)}, \dots, ex_q^{(rd)}\}$ – дополнительно назначенные выходные (праздничные) дни, признак периодичности $C = rd$;

$EX = \{\langle C_1, EX(C_1) \rangle, \langle C_2, EX(C_2) \rangle, \dots, \langle C_l, EX(C_l) \rangle\}$ – множество всех исключений для разных признаков периодичности, где l – мощность множества EX ; элементы множества EX переопределяют атрибуты периодичности для дат, указанных в $EX(C)$;

$Z = \{z_1, z_2, \dots, z_\mu\}$ – множество дней переноса расписания поездов, где z_i , $i = \overline{1, \mu}$ – это пара дней $\langle d_1^{(i)}, d_2^{(i)} \rangle$, определяющих, что расписание,

назначенное на дату $d_1^{\{i\}}$, назначается и на дату $d_2^{\{i\}}$; μ – мощность множества Z .

Сложность управления множеством версий расписания различной периодичности, постоянно появляющимися оперативными изменениями расписания, на фоне нестандартного календаря, также подверженному изменениям, потребовала разработки эффективных методов определения актуальной версии. В процессе исследования автором предложены два альтернативных метода: *метод логических правил* и *метод темпоральных элементов*, рассмотренные ниже и в работах [12, 13, 15].

10.6. Методы определения актуальной версии темпорального объекта

10.6.1. Метод темпоральных элементов

Особенность предлагаемого метода состоит в предварительном расчёте дат актуальности каждой версии расписания и сохранении их в темпоральных элементах для последующего использования.

Темпоральный элемент (ТЕ) представляет собой вычислимое множество дат и служит для определения времени, когда конкретная версия расписания поезда на протяжении своего жизненного цикла становится актуальной. ТЕ формируется с учетом данных о диапазоне действия и периодичности версии расписания, оперативных изменениях, наложениях нескольких версий и заменах.

В случае использования ТЕ версия расписания $v_i^{(n)}$ поезда с номером $n \in N$, заданная выражением (2), может быть описана выражением

$$v_i^{(n)} = (\langle n, TE_i^{(n)} \rangle, \langle s_1^{(n)}, T_1 \rangle, \langle s_2^{(n)}, T_2 \rangle, \dots, \langle s_\alpha^{(n)}, T_\alpha \rangle), \quad (3)$$

где темпоральный элемент $TE_i^{(n)}$ формируется с учётом времени t_f фиксации расписания $v_i^{(n)}$ и определяет все даты, в которые данная версия является актуальной.

Процесс вычисления ТЕ версии расписания, которая представлена кортежем отношения в форме Historical-Overlapped, включает два шага: 1) расчёт базисного значения ТЕ; 2) расчёт фактического значения ТЕ. Рассмотрим эти шаги более детально.

На первом шаге производится вычисление базисного значения темпорального элемента, обозначаемого TE^* . В расчетах учитываются только период действия версии и ее периодичность. Времена актуальности периодического события в заданном диапазоне времени $[t_s, t_e]$ представим в виде пары $\langle [t_s, t_e], C \rangle$, а множество значений TE^* в заданном диапазоне времени $[t_s, t_e]$ в виде функции ограниченного продолжения вида $BExt([t_s, t_e], C)$. Используя введённые обозначения и функции $BExt()$ и $EX()$, темпоральный элемент для периодичности C в диапазоне времени $[t_s, t_e]$ определяется следующим образом:

$$TE^* = BExt([t_s, t_e], C) \cup EX([t_s, t_e], C) - EX([t_s, t_e], \neg C), \quad (4)$$

где $EX([t_s, t_e], C) \subseteq EX(C)$ – множество дней-исключений с приобретенной периодичностью C , находящихся в период действия версии расписания, т.е. $EX([t_s, t_e], C) = \{ex^{(C)} \mid ex^{(C)} \in [t_s, t_e]\}$; $EX([t_s, t_e], \neg C)$ – множество дней-исключений, не обладающих признаком C .

Функция $BExt([t_s, t_e], C)$ определяет точки временной оси (даты) внутри диапазона $[t_s, t_e]$, отвечающие признаку периодичности C . В основе функции заложена проверка каждой точки $x \in [t_s, t_e]$ на соответствие логическому выражению C . Результат $C(x)=true$ означает, что день x удовлетворяет признаку расписания C , а $C(x)=false$ означает, что в день x действует другая версия расписания. Тогда множество точек временной оси внутри диапазона $[t_s, t_e]$, отвечающих признаку периодичности C , строится следующим образом:

$$(\forall x \in [t_s, t_e]) C(x) \Rightarrow x \in BExt([t_s, t_e], C). \quad (5)$$

В качестве примера вычисления по формуле (2) рассмотрим определение дат в диапазоне $[t_s, t_e]$, которые являются рабочими днями и чётными днями месяца. Решение этой задачи можно записать в виде:

$$(\forall x \in [t_s, t_e]) [pd(x) = true \wedge wd(x) = true] \Rightarrow x \in BExt([t_s, t_e], pd \wedge wd).$$

В общем случае при вычислении предиката $C(x)$ сначала необходимо вычислить все входящие в него предикаты $p(x)$, а затем к полученным результатам применить операции \vee, \wedge и \neg , образующие выражение C .

При определении истинности большинства высказываний $p \in P$ достаточно знания координат проверяемой точки на временной оси. Например, зная дату, можно вычислить день недели и, таким образом, проверить ее на соответствие признакам периодичности *Mon, Tue, Wed, Thu, Fri, Sat* и *Sun*, а используя правило соотношения дней недели к определению выходных дней, проверить её на соответствие признакам *wd* и *rd*. Однако существуют такие признаки, для проверки которых координат проверяемой точки недостаточно и необходимо знание о признаках периодичности других дней. Примерами такой «сложной» периодичности являются первый и последний день выходных. Для их определения необходимо знание о признаках периодичности смежных дней: день до проверяемой даты в случае первого дня выходных и день после проверяемой даты – для последнего дня выходных. Сложность состоит также в том, что для этих связанных дней могли быть назначены замены или переносы и нужно брать в расчет признаки периодичности с учётом всех этих изменений.

На втором шаге выполняется расчет фактического значения TE с использованием базисного значения TE^* . Фактическое значение темпорального элемента $TE_i^{(n)}$ версии расписания, кроме признаков периодичности и фактов переноса дат, учитывает и случаи наложения расписаний. При обнаружении таких конфликтов одинаковые даты исключаются в том из двух темпоральных

элементов, которому соответствует расписание с более ранним временем фиксации t_f . Тогда $TE_i^{(n)}$ представляет собой рекурсивную функцию. Рекурсивность заключается в том, что в вычислении темпорального элемента одной версии участвуют темпоральные элементы других более поздних версий, которые в свою очередь определяются этим же способом:

$$TE_i^{(n)} = (TE_i^* - \bigcup_{v_j^{(n)} \in V(v_i^{(n)})} TE_j^{(n)}) \cup z_i^{(-)} - z_i^{(+)}, \quad (6)$$

где $V(v_i^{(n)})$ – множество версий расписания поезда с номером n , действие которых захватывает часть периода действия версии $v_i^{(n)}$ и время их фиксации более позднее, чем у версии $v_i^{(n)}$; $z_i^{(-)}, z_i^{(+)}$ – элементы множества Z , корректирующие темпоральный элемент версии с учётом замен, вида:

$z_i^{(-)} = \{d_1 \mid z = \langle d_1, d_2 \rangle, z \in Z, d_2 \in TE_i^{(n)}\}$ – множество дат, которые должны быть исключены из темпорального элемента версии $v_i^{(n)}$, так как на них назначена замена;

$z_i^{(+)} = \{d_1 \mid z = \langle d_1, d_2 \rangle, z \in Z, d_1 \in TE_i^{(n)}\}$ – множество дат, которые должны быть включены в темпоральный элемент версии $v_i^{(n)}$, так как ее темпоральный элемент $TE_i^{(n)}$ содержит дату, расписание которой является эталоном для даты переноса.

Отметим, что после коррекции по формуле (6) темпоральные элементы различных версий расписания поезда n не пересекаются, т.е.

$$TE_i^{(n)} \cap TE_j^{(n)} = \emptyset \text{ для } i \neq j; i, j = 1, 2, \dots, g.$$

На рис. 14 проиллюстрирован перенос расписания с пятницы 13 апреля на субботу 14 апреля, заключающийся в исключении 14 апреля из темпорального элемента расписания #2 для выходных дней (rd) и включении его в темпоральный элемент расписания #1 для рабочих дней (wd).

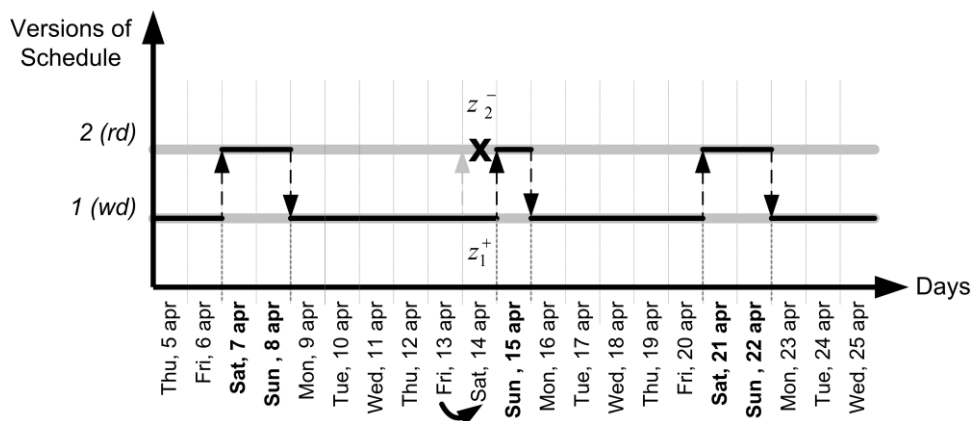


Рис. 14. Перенос расписания с одного дня на другой

Реализация метода построения ТЕ с использованием различных подходов рассмотрена в 3-й главе диссертации и в работах [12, 13, 15].

Наличие ТЕ позволяет значительно упростить многие темпоральные функции, такие как определение актуальной версии объекта, определение точек на оси действительного времени, когда конкретная версия объекта выходит из теневой зоны. Для *определения актуальной версии* расписания поезда с номером n предлагается использовать функцию $ActVer()$ вида:

$$ActVer(x, V^{(n)}) = \begin{cases} v_i^{(n)}, & \text{if } x \in TE_i^{(n)}; \\ null, & \text{otherwise,} \end{cases} \quad (7)$$

где x – дата, на которую необходимо знать расписание.

Результатом функции $ActVer(x, V^{(n)})$ является элемент множества $v_i^{(n)} \in V^{(n)}$, представляющий собой искомую актуальную версию. В случае если поезд не ходит в указанную дату x , функция вернет неопределенное значение ($null$). Гарантом существования только одной актуальной версии является условие непересечения ТЕ всех версий расписания поезда.

Одной из серьёзных проблем использования ТЕ является *поддержка множества ТЕ* в актуальном состоянии, так как в случае появления изменений расписания темпоральные элементы версий могут стать устаревшими и их необходимо пересчитывать. Однако многие поезда не прекращают курсирования по несколько лет, имеют при этом множество версий расписания, часть из которых относится только к прошлому, и вводимые изменения, затрагивающие настоящее и будущее, никаким образом на них не отражаются. То есть затронутый изменениями диапазон может не перекрываться с жизненным циклом версии или перекрываться только частично. В таком случае нецелесообразно обновлять ТЕ всех версий расписания поезда, а также пересчитывать их целиком. Разработанный автором метод частичного пересчета ТЕ представлен в 3-й главе диссертации.

10.6.2. Метод логических правил

Суть рассматриваемого метода состоит в вычислении актуальной версии расписания на основе логических правил, которые учитывают все особенности расписания: многоверсионность, вызванную оперативными изменениями, многовариантность в случае периодического расписания, замены и переносы. Метод работает с данными темпоральной базы данных расписания и сразу же учитывает все изменения в расписании, как только они были введены в систему.

Расчёты метода используют множество версий расписания $V^{(n)}$ поезда с номером n , множество исключений EX и множество замен Z . Версия

расписания поезда представлена согласно формуле (2), в расчётах используется только её идентификационная часть: $\langle n, C, t_s, t_e, t_f \rangle$.

Центральным параметром метода, относительно которого выполняются все логические правила, заложенные в алгоритм, является день x – дата, для которой нужно определить актуальную версию расписания поезда. В связи с существованием особого календаря необходимо проверить дату x на предмет переопределения признаков периодичности и замен. Если таковые найдены, то дату x необходимо переопределить в дату \hat{x} и дальнейшие расчеты проводить относительно \hat{x} :

$$\hat{x} = \begin{cases} d_1^{\{i\}}, & \text{if } \exists z_i \in Z \mid z_i = \langle d_1^{\{i\}}, d_2^{\{i\}} \rangle, d_2^{\{i\}} = x; \\ x, & \text{otherwise.} \end{cases} \quad (8)$$

На дату \hat{x} действительными могут быть только те версии расписания, у которых, во-первых, диапазон действия покрывает эту дату, а во-вторых, признак периодичности удовлетворяет этой дате. Следовательно, необходимо среди множества всех версий расписания поезда n выделить версии расписания, подходящие по времени действия и по периодичности, то есть только те, которые действуют в день \hat{x} и соответствуют ему по признаку C . Версией, соответствующей по периодичности дню \hat{x} , является та версия, для которой множество истинности её предиката $C(\hat{x})$ на области действия этой версии $[t_s, t_e]$ включает дату \hat{x} , т.е. $\hat{x} \in \{y \mid y \in [t_s, t_e], C(y)\}$. Однако совсем не обязательно вычислять множество истинности, достаточно само высказывание C применить ко дню \hat{x} . Если результат будет истинен, то версия удовлетворяет периодичности дня \hat{x} . Таким образом, множество всех версий, подходящих по времени действия и периодичности дате x выделяется из множества версий расписания поезда $V^{(n)}$ согласно следующей формуле:

$$V_{\hat{x}}^{(n)} = \{v \mid v \in V^{(n)}, \hat{x} \in [t_s, t_e], C(\hat{x})\}. \quad (9)$$

Для учёта исключений в периодичности в высказывании C предикаты $p_i(\hat{x})$ заменяются на $\hat{p}_i(\hat{x})$. Предикат $\hat{p}_i(\hat{x})$ по сути соответствует описанному в п.10.6.1. предикату $p_i(\hat{x})$, но предваряет его вычисление дополнительной проверкой на наличие исключений. В случае обнаружения таковых дате \hat{x} присваивается признак p_i . Следующая формула это показывает:

$$\hat{p}_i(\hat{x}) = \begin{cases} True, & \text{if } \hat{x} \in EX(C), EX(C) \in EX, p_i = C; \\ p_i(\hat{x}), & \text{otherwise.} \end{cases} \quad (10)$$

Примечание. Если в системе расписания используются признаки периодичности, зависящие от признаков периодичности смежных дат, такие как lh – «последний выходной день недели» и fw – «первый рабочий день недели», то предикатные формулы $p_i(\hat{x})$ и $\hat{p}_i(\hat{x})$, участвующие в S , необходимо расширить до $p_i(x, \hat{x})$ и $\hat{p}_i(x, \hat{x})$ соответственно. Невозможно корректно определить признаки периодичности смежных дат, имея информацию только о \hat{x} .

В случае существования пересекающихся версий расписания (форма Historical-Overlapped) множество $V_{\hat{x}}^{(n)}$ может содержать более одного элемента. Следовательно, из множества $V_{\hat{x}}^{(n)}$ необходимо выделить тот единственный элемент, который и будет являться актуальной версией. При расчете приоритет отдаётся той версии, которая позже всех была введена в систему:

$$v_a^{(n)}(x) = (v / v \in V_{\hat{x}}^{(n)}, t_f = \max_{V_{\hat{x}}^{(n)}} \{t_f\}). \quad (11)$$

Достоинством рассмотренного подхода является то, что все вычисления происходят с оперативными данными напрямую, таким образом, любые изменения после их внесения в систему сразу учитываются в запросах к расписанию. В качестве недостатка метода нужно отметить то, что каждый раз для выполнения любого запроса к системе, в котором необходимо знание актуальной версии расписания поезда, алгоритм поиска, состоящий из множества операций, приходится выполнять заново.

В работе проведена апробация предложенных методов определения актуальной версии объекта в ИС расписания движения поездов на Латвийской железной дороге и определена эффективность их применения при решении различных задач, возникающих на практике. Процедура данного исследования включала:

1. Выделение функциональных задач ИС, в которых определяются актуальные версии объекта, выбор задач подлежащих рассмотрению;
2. Формирование системы показателей эффективности методов определения актуальной версии объекта и выбор способа их оценивания,
3. Апробация методов при решении выбранных задач ИС, оценка их эффективности и разработка рекомендаций по их практическому применению.

Результаты оценки сравнительной эффективности и возможности применения методов определения актуальной версии объекта в ИС расписания движения поездов представлены в табл.2. Каждый из показателей принимает одно из следующих двух значений: «+» – если по данному показателю метод превосходит своего конкурента; «-» – в противном случае.

Таблица 2. Сравнение методов идентификации актуальной версии объекта

	Метод ЛП	Метод ТЕ
Показатели эффективности	Оценка сравнительной эффективности методов	
Необходимость предварительной подготовки данных к использованию в запросах после изменений в расписании	+	–
Сложность поиска актуальной версии (количество операций при поиске) *	+ (> 7)	+ (1)
Быстродействие поиска актуальной версии **	–	+
Объем обрабатываемых данных при поиске актуальной версии	–	+
Необходимость выполнения ограничений временной области для запросов	+	–
Оперативность внесения изменений	+	–
Обеспечение целостности данных при изменении расписания	+	–
Объем хранимых данных ***	+ (~27КВ)	– (~2000КВ)
Прикладные задачи системы расписания	Оценка возможности применения	
Поиск актуальной версии расписания поезда на заданный день	+	+
Построение последовательности смены версий на заданный диапазон	– (для больших диапазонов) + (для небольших диапазонов)	+
Поиск интервалов активности конкретной версии	–	+
Поиск ближайшего изменения в расписании	–	+
Выявление временных точек (дат) смен расписаний в заданном диапазоне	–	+

Поясним показатели, приведенные в табл. 2.

* *Сложность поиска* актуальной версии оценивалась количеством выполняемых операций. В методе ЛП требуется применение серии логических правил, которые в зависимости от количества версий и их признака C будут состоять как минимум из семи логических проверок. Среднее количество операций можно подсчитать следующим образом: $7 + |\overline{V^{(n)}}| \cdot \overline{C}$, где $|\overline{V^{(n)}}|$ – среднее количество версий расписания одного поезда (мощность множества $\overline{V^{(n)}}$), \overline{C} – средняя местность высказывания C (среднее количество признаков в C). Для расписания Латвийской железной дороги среднее количество операций равняется 15 операциям (в 2004 году значение $|\overline{V^{(n)}}|$ составляло 6 версий, в 2005 г. – 4, 2006 г. – 3, 2007 г. – 4, при этом $\overline{C} = 1.8$). В методе ТЕ поиск представляет собой всего одну элементарную операцию проверки вхождения в темпоральные элементы.

** *Быстродействие*. Время поиска зависит от особенностей реализации, СУБД и так далее. Значение этого показателя неизбежно коррелирует с количеством логических проверок. Поэтому по данной позиции метод ТЕ безусловно выигрывает.

*** Объем хранимых данных сравнивался по идентификационной части расписания, то есть без учёта данных об остановках на станциях, так как объем этих данных для обоих методов одинаков. Сравнение производилось на базе фактического расписания пассажирских поездов Латвийской железной дороги на 2004 год: 389 поездов, 2031 версия расписания.

10.7. Метод контроля доступа к темпоральным данным на уровне кортежей отношения

Особенность защиты темпоральных данных состоит в том, что данные различных временных периодов логически располагаются в одном месте – в едином реляционном отношении. Разграничение доступа пользователей к данным различных временных периодов, то есть к подмножеству кортежей реляционного отношения, не входит в набор стандартных средств обеспечения безопасности реляционной СУБД, что потребовало дополнительных разработок.

Предложенный автором метод основывается на ограничении доступа к данным конкретных строк таблицы для операций чтения, модификации и удаления. Применяя этот метод, можно ограничить доступ таким образом, чтобы пользователь к одной записи таблицы (версии) смог получить доступ, а к другой – нет. В реализации метода используются только стандартные механизмы SQL-2: триггеры и представления, математический аппарат системы – реляционная алгебра. Метод рассмотрен и применён в 3-й главе диссертации и работах автора [4, 7, 9].

Для формализации задачи определим множества, участвующие в процессе контроля доступа:

$\mathfrak{R} = \{R_1, R_2, \dots, R_n\}$ – множество отношений БД, $|\mathfrak{R}| = n$, где $R_i, i = \overline{1, n}$ имеет схему $R_i(A, L)$;

r – кортеж отношения R_i , объект защиты, $r \in R_i$;

$A = \{a_1, a_2, \dots, a_k\}$ – подмножество атрибутов R_i , не влияющих на доступ к его кортежам, $|A| = k$;

$L = \{l_1, l_2, \dots, l_m\}$ – подмножество атрибутов R_i , определяющих доступ к его кортежам, $|L| = m$. Среди L могут быть темпоральные атрибуты, специализированные атрибуты безопасности или другие атрибуты, а также их скалярные производные;

$U = \{u_1, u_2, \dots, u_z\}$ – множество зарегистрированных пользователей системы, $|U| = z$;

$\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4\}$ – множество операций доступа к кортежу, $|\Theta| = 4$, $\Theta = \{\text{SELECT, INSERT, UPDATE, DELETE}\}$;

$S = \{S_1, S_2, \dots, S_h\}$ – множество наборов правил политики безопасности, регламентирующих доступ пользователей к данным отношений из \mathfrak{R} , $|S| = h$. Наборы правил S_1, S_2, \dots, S_h могут иметь разную структуру в зависимости от

конкретного правила политики безопасности. Множество $S_j \in S$, $j = \overline{1, h}$, содержит экземпляры правил $s_1^{(j)}, s_2^{(j)}, \dots, s_f^{(j)}$ одинаковой структуры, $|S_j| = f$. Такая организация правил позволяет при реализации метода размещать их в едином реляционном отношении в виде набора кортежей, манипулируя которыми можно оперативно управлять доступом пользователей к данным. К примеру, для ограничения доступа к темпоральным данным отношения R_i в зависимости от их расположения на временной оси правило $s_y^{(j)} \in S_j$, будет иметь вид:

$$s_y^{(j)} = \langle R_i, u_c, t_s, t_e, \theta_q, g_d \rangle, j = \overline{1, h}. \quad (12)$$

Правило $s_y^{(j)}$ запрещает ($g_d=0$) или разрешает ($g_d=1$) пользователю $u_c \in U$ выполнение операции $\theta_q \in \Theta$ над версией объекта, представленной кортежем r отношения $R_i \in \mathfrak{R}$, и актуальной в диапазоне времени $[t_s, t_e]$.

Вычисление факта доступности кортежа r определяется в момент запроса к нему с помощью предиката $P_b \in \Pi$, где $\Pi = \{P_1, P_2, \dots, P_h\}$ – множество предикатов системы безопасности ИС, использующее для вычислений множество S ; $|\Pi| = h$, $b = \overline{1, h}$. В общем виде каждый предикат имеет структуру:

$$P_b(R_i, u_c, \theta_q, L, t), \quad (13)$$

где R_i – отношение, к которому относится кортеж r ; u_c – пользователь, осуществляющий операцию θ_q над кортежем r ; t – время запроса (операции); L – значения атрибутов кортежа отношения R_i , влияющие на разрешение доступа. Для вышеприведенного примера ограничения доступа по временному диапазону $L = (t_s, t_e)$. В случае истинности P_b доступ к кортежу $r = (A, L)$ разрешается, в противном случае ограничивается.

Принудительное применение предикатов из Π реализуется в представлениях данных (View), используемых для осуществления запросов на чтение и в триггерах, контролирующих запросы на модификацию данных.

В 3-й главе диссертации рассмотрен пример применения метода для обеспечения конфиденциальности данных темпоральной таблицы железнодорожных тарифов.

При разработке метода ограничения доступа учитывалось, что система безопасности данных с учётом их темпорального измерения должна интегрироваться с действующей системой безопасности БД, так как темпоральное измерение – это не единственный параметр, по которому ограничивается доступ к данным ИС, функционирующих на транспорте. Поэтому предложенный автором метод защиты данных позволяет дифференцировать доступ не только относительно темпорального измерения данных, но также и любого другого присутствующего в данных свойства:

территориального или географического, административного и тому подобного. Разработке требований к комплексной системе безопасности реляционных данных посвящена работа автора [7].

10.8. Метод трансформации интервальных данных

Используемые в транспортных ИС модели представления многоверсионных данных хранят версии, как правило, в интервальной форме, которая неудобна для проведения аналитических расчетов. Так большинство используемых методов анализа данных, и, соответственно, аналитических систем, оперирует не с интервальными формами представления данных, а с точечными (например, временными рядами). В точечной форме факт ассоциируется с множеством моментов времени выбранной гранулярности (например, с днями, месяцами или годами). Операции над такими множествами очень удобны для теоретических выкладок и формализации. Однако, математический аппарат реляционных СУБД не предоставляет готовых средств для преобразования интервальной формы в точечную, что потребовало разработки соответствующих механизмов в настоящем исследовании.

В рамках решения поставленной задачи была разработана общая схема трансформации интервальных темпоральных данных, показанная на рис. 15.

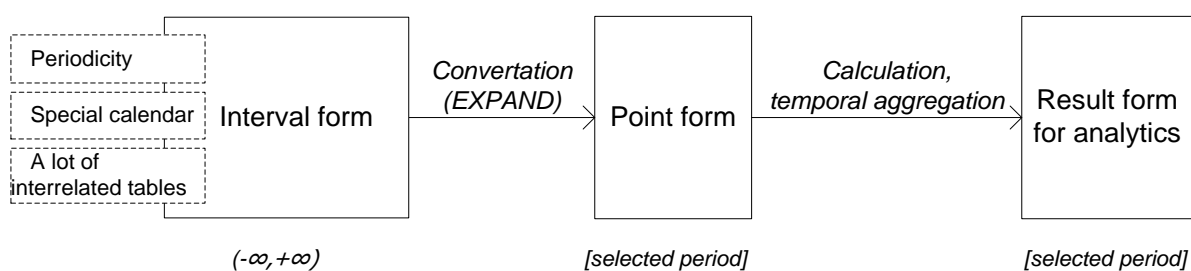


Рис. 15. Схема трансформации темпоральных данных

Поясним представленную схему.

Классическая интервальная форма (*Interval form*) изначально неудобна для ряда вычислений, но их сложность многократно возрастает, когда присутствуют следующие аспекты: периодичность, особый календарь и необходимость использования в вычислениях данных нескольких взаимосвязанных таблиц⁵.

Применение предложенной автором операции EXPAND к темпоральным данным интервальной формы решает проблему «неудобного» для вычислений представления данных, превращая интервальный факт во временной ряд, где каждое значение характеризует существование этого факта в определенный момент времени (*Point form*). Затем данные подвергаются дальнейшей обработке, которая часто связана с операциями темпоральной агрегации. На этом этапе могут быть и более серьезные вычисления, связанные с

⁵ Объекты ИС в базах данных согласно реляционной теории представляются в виде взаимосвязанных таблиц. Поэтому получение требуемого результата, как правило, требует обработки нескольких таблиц.

отображением темпоральных концептов. Конечным результатом обработки является требуемая форма данных (*Result form for analytic*).

Исходные множества для каждого этапа охватывают различные периоды времени. Множество темпоральных данных интервальной формы (первый блок на рис. 15) может покрывать неограниченный временной диапазон $(-\infty, +\infty)$. С переходом к точечной форме (второй и третий блоки) временная область темпоральных данных сужается до области анализа (*selected period*), что сильно сокращает объём данных в вычислениях и существенно экономит вычислительные ресурсы, а также решает проблему существования версий с неопределёнными значениями начала или завершения периода действия.

Рассмотрим предлагаемый метод трансформации интервальных данных в точечную форму. Имеется два отношения:

$R1$ – темпоральное отношение интервальной формы, описывающее множество версий объекта и имеющее схему $R1(A, t_s, t_e, C)$, где $A = \{a_1, a_2 \dots a_k\}$ – множество атрибутов, описывающих объект, k – количество атрибутов. Каждая из версий имеет период действия от момента t_s до момента t_e и обладает признаком периодичности C . Моменты актуальности версии, кроме того, зависят от особого календаря Z , который описывает переносы признаков одних дней на другие (дополнительные выходные или рабочие дни и т.д.);

$R2$ – темпоральное отношение точечной формы, имеющее схему $R2(A, d)$, где d – момент времени, в который факт A является актуальным.

Необходимо преобразовать $R1(A, t_s, t_e, C)$ в $R2(A, d)$, при этом набор данных $R2$ должен ограничиваться фиксированным отрезком времени (заданным периодом анализа) $T^{(q)} = [t_s^{(q)}, t_e^{(q)}]$, где $\forall d \in T^{(q)}$.

Формула трансформации выглядит следующим образом:

$$R2 = \sigma_{d \in T^{(q)}} R1 \times \tau(t_s, t_e, C), \quad (14)$$

где $\tau(t_s, t_e, C)$ – функция, раскладывающая диапазон $[t_s, t_e]$ на множество дат d , отвечающих заданной периодичности C :

$$\tau(t_s, t_e, C) = \{d \mid d \in [t_s, t_e], p(d, C) = true\}, \quad (15)$$

$p(d, C)$ – предикат периодичности, истинен в случае, если дата d имеет признак C . Предикат учитывает специальный календарь Z (к примеру, перенос расписания выходных дней на пятницу и понедельник в связи с национальными праздниками в Латвии 1 мая и 4 мая в 2009 году);

$R1 \times \tau(t_s, t_e, C)$ – конструкция, реализующая операцию развёртывания EXPAND, которая является центральной операцией всего преобразования и обеспечивает переход от интервальной формы к точечной. Селекция $\sigma_{d \in T^{(q)}}(\dots)$ ограничивает результат областью анализа $T^{(q)}$. Мощность результирующего множества может достигать $|R1| \cdot |\tau(T^{(q)})|$.

После выполнения операции развёртывания дальнейшие преобразования (группировка, естественное соединение, проекция) становятся тривиальными.

Использование предложенной функции $\tau(t_s, t_e, C)$ позволяет максимально упростить вычисления. Для её реализации используется механизм рекурсии и на языке SQL операция формирования множества дат $D = \{d\}$ заданной периодичности C между моментами времени $dStart$ и $dEnd$ может быть представлена следующим образом:

```
WITH D (d) AS
( VALUES dStart
  UNION
  SELECT d + 1 day as d FROM D WHERE d < dEnd )
SELECT d FROM D WHERE p(d, C).
```

10.9. Применение результатов исследования в системах хранилищ данных

В ходе данного исследования разработан концептуальный подход к хранению и анализу корпоративных данных в ИС Латвийской железной дороги, основанный на технологии хранилищ данных, который реализован на практике и опубликован в работах [1, 2, 8]. В этих работах автор решает ряд практически важных задач, среди которых отметим построение системы обработки больших объемов данных, разработку принципов построения хранилищ данных в системах поддержки принятия решений на железной дороге, обеспечение целостности и безопасности хранимых данных и др. При этом особое внимание уделено проблеме использования в хранилищах темпоральных данных. Последнее объясняется тем, что основным измерением в хранилищах данных является время. С ним связаны основные процессы функционирования хранилища данных: выгрузка, трансформация, загрузка (Extract, Transform, Load, т.е. сокращённо ETL), дальнейшая обработка данных и создание витрин. Организация в хранилищах эффективного взаимодействия с временным измерением данных определяет безопасность и целостность данных, качество их обработки (количество ошибок), простоту реализации процессов функционирования хранилищ данных и оперативность внесения в них изменений.

Основная проблема взаимодействия с темпоральными данными заключается в том, что оператор или используемые программные средства могут ошибочно взаимодействовать с данными другой временной зоны, не относящейся к обрабатываемому временному диапазону. Это объясняется несовершенством механизмов работы с темпоральным измерением данных, а в некоторых случаях полным отсутствием этого измерения в используемых БД. Последнее относится прежде всего к базе данных НСИ. Данная проблема имеет место на разных этапах обработки данных в хранилище. Технологический процесс подготовки данных для аналитической системы пассажирских перевозок на Латвийской железной дороге схематически представлен на рис. 16. Данные из оперативных источников (*Data Sources*) проходят декодирование и очистку, затем они погружаются в хранилище (*Data Warehouse*) и распределяются по витринам данных (*Data Marts*). Распределение является сложным процессом трансформации, в котором участвует база данных

НСИ, содержащая зависимые от времени правила трансформации данных. После загрузки данных в витрины аналитик оценивает результат и принимает решение о готовности данных для дальнейшего анализа и генерации отчетов. При оценке результата аналитик руководствуется агрегированными данными из оперативных систем и другими источниками. Если результат неудовлетворителен, то часть технологического процесса повторяется и дополняется действиями по очистке хранилища и витрин. В зависимости от источника возникновения ошибки технологический процесс возобновляется с различных мест. Источник ошибки может быть в процедурах декодирования и очистки, в исходных данных, в базе данных НСИ и в процессах трансформации данных.

Критичность вышеописанных операций повышается при нарушении принятого порядка загрузки данных в аналитическую систему, когда актуальный обрабатываемый период находится дальше в прошлом, чем ранее обработанный период. В этом случае автором выявлены следующие проблемы:

- трансформация данных для витрин основывается на данных предыдущего обрабатываемого периода, и следовательно, часть технологического процесса должна выполняться циклично, начиная с обрабатываемого периода вплоть до последнего, находящегося в системе;
- процедуры очистки витрин и трансформации данных не предназначены для обработки данных за любой период времени и работают в том диапазоне времени, который определен установленным порядком их использования;
- НСИ (при нетемпоральной БД или неверно взятом её темпоральном срезе) не соответствует обрабатываемому периоду, поэтому результат обработки данных может быть некорректен.

В технологическом процессе функционирования информационно-аналитической системы выявлены, как минимум, 4 критические операции, при выполнении которых вероятность нарушения целостности информационных ресурсов хранилища высока. К ним относятся:

- частичное удаление данных в хранилище;
- исправление данных в хранилище;
- корректировка базы данных НСИ;
- частичное удаление данных в витринах.

Таким образом, качество функционирования данной системы напрямую зависит от наличия:

- темпорального измерения в базе данных НСИ;
- простых и удобных средств для получения темпоральных срезов (таких, которые бы не сильно усложняли код обработки данных);
- средств настройки на обрабатываемый период для облегчения процедур обработки данных, а также предотвращения непреднамеренного изменения данных другого периода.

Уровень накопления данных, содержащихся в информационном объекте

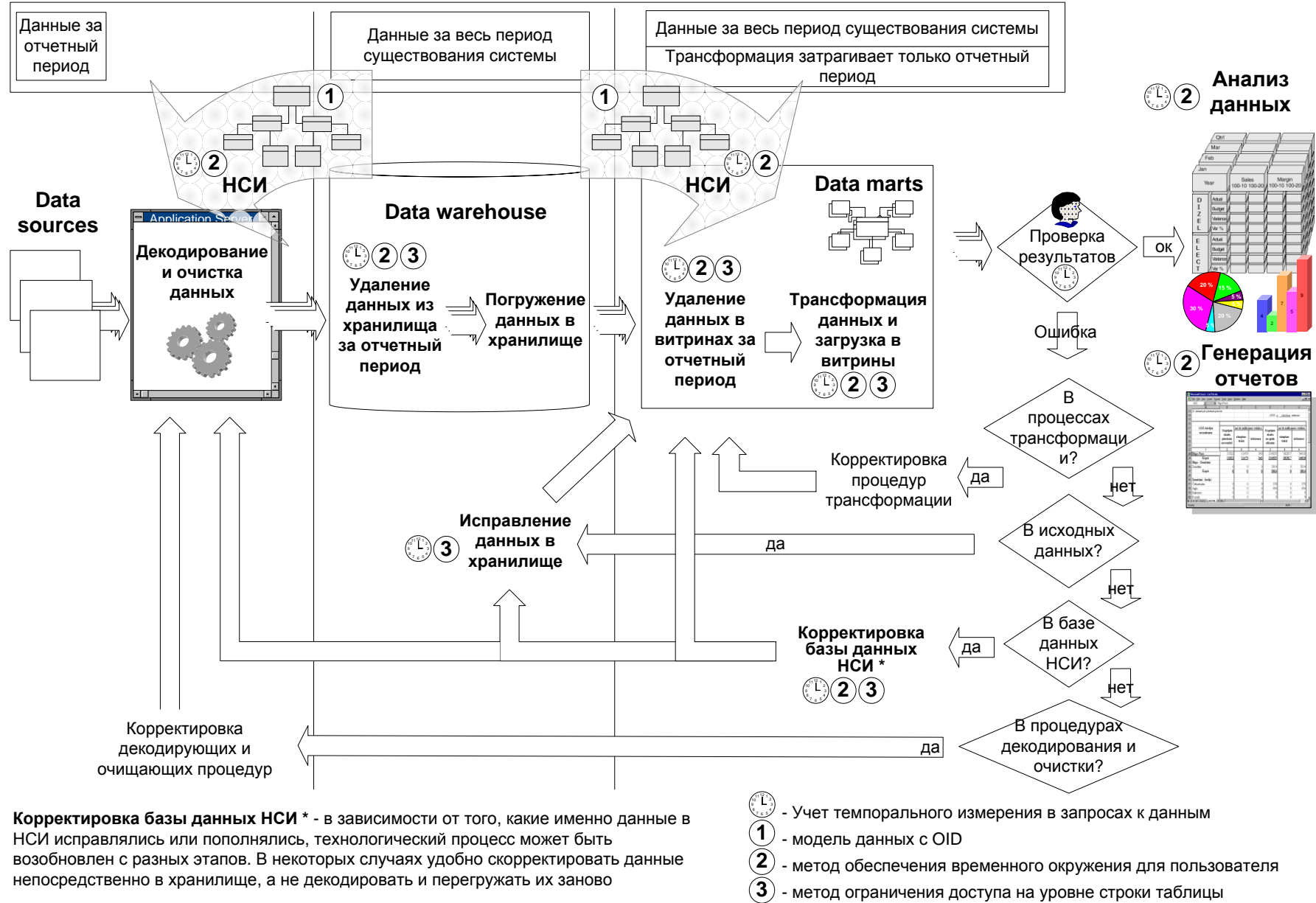


Рис. 16. Схема пополнения данных хранилища и подготовки данных к анализу

Применение разработанных в диссертации методов и моделей позволяет решить указанные проблемы, а именно: снизить риск нарушения целостности данных хранилища, упростить трансформацию и подготовку данных к анализу за любой период времени, повысить оперативность и безопасность этих операций:

1. Для решения задачи попадания в определенный временной срез, соответствующий обрабатываемому периоду предложено использовать *метод установки временного окружения пользователя*. Эта задача востребована на многих этапах технологического процесса, начиная от этапа декодирования, заканчивая этапом трансформации данных в витрины и проверкой результатов.

2. Для ограничения доступа к данным хранилища, относящимся к необрабатываемым диапазонам времени, с целью предотвращения их непреднамеренного изменения, предложено использовать *метод разграничения доступа к данным реляционных систем на уровне строк отношения* [4, 7, 9].

3. *Метод разграничения доступа к данным реляционных систем на уровне строк отношения* предложено использовать и при изменении базы данных НСИ.

4. Так как база данных НСИ является темпоральной, то процессы изменения данных пользователем, находящимся в определенном временном срезе, будут нетривиальными. Если оператор специально не определяет значения *временных* атрибутов для измененных данных, то появляющаяся при изменении данных новая версия объекта будет иметь время действия равное временному диапазону пользователя, описанному в его *временном окружении*. Таким образом, изменения в НСИ не затронут другие временные периоды.

На схеме на рис. 16 пометками ⌚ и ①, ②, ③ показаны области применения разработанных моделей и методов при обработке темпоральных данных, среди них: запросы к темпоральным данным НСИ, темпоральная выборка и агрегация, удаление данных определенного временного периода, ограничение доступа к данным, относящимся к необрабатываемому временному диапазону.

10.10. Применение результатов исследования в интерактивной информативно-справочной системе расписания пассажирских поездов

Система интерактивного расписания пассажирских поездов SAR является одним из основных объектов данного исследования. Объект интересен тем, что оперирует с темпоральными данными, которые обладают максимальным количеством темпоральных особенностей по сравнению с другими ИС предприятия Латвийской железной дороги. Среди них выделяем многоверсионность, неоднократное вступление в силу одной и той же версии, периодичность, особый календарь. Добавим к этому большой объем данных и требование высокого быстродействия при запросах к этим данным.

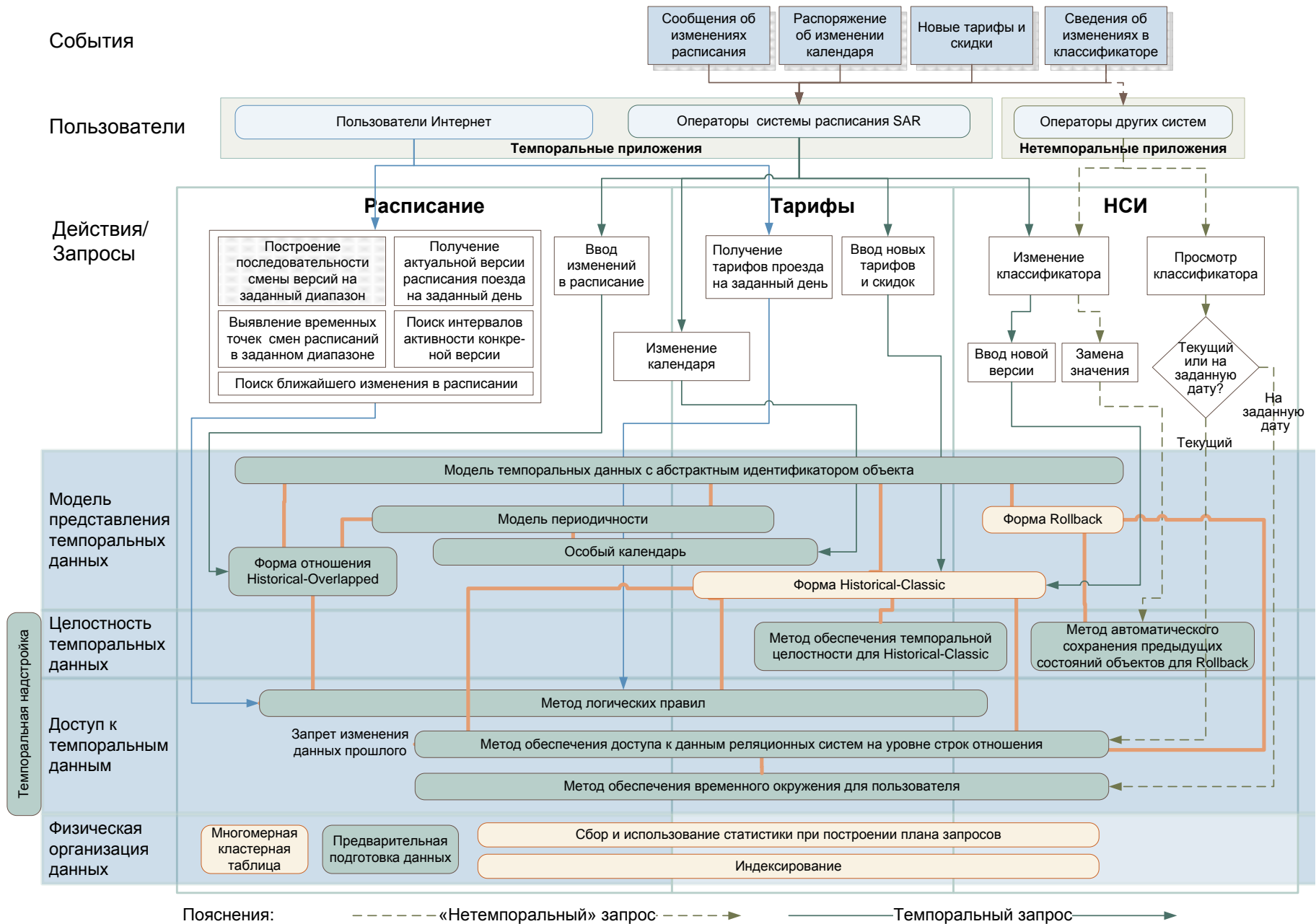


Рис. 17. Схема применения разработанных методов и моделей в системе интерактивного расписания пассажирских поездов

Система SAR имеет WEB интерфейс, через который все пользователи Интернет могут в интерактивном режиме запрашивать расписание пассажирских поездов на любой день и направление. Система позволяет учитывать оперативные изменения в расписании и планировать расписание на длительный период.

Система оперирует с данными, предметно относящимися к трём областям: расписание, тарифы и классификаторы (НСИ). Пользователей, взаимодействующих с данными этой системы, по характеру запросов можно разделить на 3 группы: пользователи Интернет, операторы систем ввода расписания, тарифов и операторы других систем, среди которых есть и унаследованные системы, не «умеющие» работать с темпоральными данными.

Схема применения разработанных методов и моделей в системе интерактивного расписания пассажирских поездов при взаимодействии различных категорий пользователей с данными системы расписания показана на рис. 17.

Методы и модели разделены на 4 группы: модель представления темпоральных данных, целостность темпоральных данных, доступ к темпоральным данным и физическая организация данных.

Использование разработанных методов и моделей в организации системы SAR позволило:

1. Существенно упростить разработку системы;
2. Организовать данные расписания и тарифов наиболее компактным образом, учитывая все перечисленные выше темпоральные особенности данных (модель AOID, форма Historical-Classic, форма Historical-Overlapped, модель периодичности и особого календаря);
3. Преобразовать фрагменты централизованной базы данных классификаторов в темпоральные, при этом обеспечив темпоральную восходящую совместимость с другими ранее разработанными приложениями (форма Rollback, форма Historical-Classic, метод автоматического сохранения предыдущих состояний объектов, метод обеспечения временного окружения для пользователя);
4. Обеспечить корректный доступ к актуальным версиям данных (метод логических правил);
5. Обеспечить темпоральную целостность данных (метод обеспечения темпоральной целостности для Historical-Classic, метод автоматического сохранения предыдущих состояний объектов);
6. Обеспечить неприкосновенность истории данных расписания, тарифов и классификатора (метод ограничения доступа к данным реляционных систем на уровне строк отношения);
7. Достигнуть с помощью использования различных методик индексирования, в том числе и кластеризации требуемого быстродействия;

В данный момент система SAR, реализованная с использованием разработанных методов и моделей, успешно применяется на Латвийской железной дороге.

ЗАКЛЮЧЕНИЕ

Основные результаты проведённых исследований могут быть сформулированы в виде следующих выводов:

1. Определены основные группы темпоральных данных в информационных системах предприятия ГАО «Латвийская железная дорога», рассмотрена их взаимосвязь с решаемыми задачами. Выявлены проблемы использования темпоральных данных в действующих ИС, подлежащие решению, в их числе: несовершенство моделей темпоральных объектов; неэффективная поддержка целостности данных; сложности обеспечения конфиденциальности данных; отсутствие аппарата эффективной обработки темпоральных данных; несовершенство процессов подготовки данных к анализу.

2. Разработан ряд теоретических положений в области темпоральных БД, в их числе: введение новой формы темпорального отношения *Historical-Overlapped* (отношение с пересекающимися временными отрезками); создание оригинальной классификация моделей и методов, подлежащих к использованию в рассматриваемых темпоральных системах; формализация ограничений темпоральной целостности для темпоральных форм отношений (условия непересечения временных отрезков).

3. Предложены принципы построения темпоральных баз данных в реляционной среде, использующие модель темпоральных данных с абстрактным идентификатором объекта и темпоральную логику.

4. Разработаны оригинальные модели и методы для работы с темпоральными объектами с учётом специфики задач на железнодорожном транспорте:

- модель темпорального отношения с пересекающимися жизненными отрезками;
- модель темпорального объекта, обладающего периодичностью и особым календарём;
- метод обеспечения временного окружения для пользователя;
- методы определения актуальной версии темпорального объекта, в их числе: метод логических правил и метод темпоральных элементов;
- метод трансформации интервальных данных;
- метод защиты темпоральных данных, учитывающий повышенные требования железнодорожных ИС к целостности, конфиденциальности и доступности данных.

5. Корректность разработанных моделей и методов проверена при решении разнообразных задач информационно-аналитических систем ЛЖД, в первую очередь, в задаче расписания поездов. Полученные результаты подтверждают валидность представленных моделей и применимость методов.

6. Результаты исследования позволили разработать для Латвийского предприятия ГАО «Латвийская железная дорога» интерактивную ИС

расписания пассажирских поездов, повысить безопасность темпоральных данных в системе обработки пассажирской отчетности и модернизировать централизованную БД железнодорожных классификаторов и кодификаторов. Предложенная система поддержки расписания может служить не только основой ИС железнодорожных перевозок, но и использоваться как часть системы управления, позволяющей моделировать расписание поездов (автоматически генерируя рекомендации аналитику для переноса поездов на другой день недели при объявлении нового выходного дня) и как часть аналитической системы для анализа и прогнозирования пассажиропотоков.

7. Результаты исследований апробированы в ИС предприятия ГАО «Латвийская железная дорога», однако они могут использоваться в ИС различных железнодорожных компаний. Более того во многих случаях полученные результаты имеют универсальный характер и могут применяться в темпоральных ИС и в других областях.

8. Результаты исследования используются в процессе обучения студентов Института транспорта и связи в курсах «Базы и банки данных» бакалаврской программы «Компьютерные науки» и в курсе «Современные технологии баз данных» магистерской программы «Компьютерные науки».

СПИСОК ОПУБЛИКОВАННЫХ АВТОРОМ РАБОТ

Статьи

1. Копытов Е.А., Петухова Н.Ю., Демидов В.В. Методология построения системы обработки больших объёмов данных для анализа пассажирских перевозок на Латвийской железной дороге. В кн.: *Proceedings of VI International Conference "TransBaltica 2001", June 7-8, 2001, Riga, Latvia*. Riga: RMS Forum, 2001, с. 201-208.
2. Петухова Н. Методология построения системы обработки больших объёмов данных. В кн.: *Informācijas tehnoloģija: zinības un prakse. VI konference. Konferenču materiāli. 2001.gada 28.novembrī. Rīga, Latvija*. Riga: DATI, 2001, с. 65-70.
3. Копытов Е.А., Демидов В.В., Петухова Н.Ю. Использование принципов временных баз данных в информационных системах на Латвийской железной дороге. В кн.: *Proceedings of VII International Conference "TransBaltica 2002", June 12-14, 2002, Riga, Latvia*. Riga: RMS Forum, 2002, с. 183-190.
4. Петухова Н. Метод обеспечения доступа к данным реляционных систем на уровне строк отношения. *Transport and Telecommunication*. Vol.4, No 1, 2003, с. 45-52.
5. Kopitovs J., Demidovs V., Petukhova N. Method of Temporal Databases design Using Relational Environment. In: *Scientific proceedings of Riga Technical*

University - Computer Science: Applied Computer Systems, Series #5, Issue #13. Riga: RTU, 2002. pp. 236-246.

6. Kopitovs J., Demidovs V., Petukhova N. Virtual Models in Forecasting Systems of Railway Transportation. In: *The international conference: Modelling and Simulation of Business Systems (MOSIBUS 2003). May 13-14, 2003. Vilnius, Lithuania*. Kaunas: Kaunas University of Technology Press, 2003, pp. 265-268.
7. Петухова Н. Принципы разработки комплексных систем безопасности в реляционных базах данных на примере Латвийской железной дороги. *Transport and Telecommunication*, Vol.5, No 2, 2004, с. 36-44.
8. Kopytov E., Demidovs V., Petoukhova N. Principles of Creating Data Warehouses in Decision Support Systems of Railway Transport. In: *Computing Anticipatory Systems: CASYS 2003 - Sixth International Conference, Liège, Belgium, 11-16 August 2003. AIP Conference Proceedings*. Melville, New York: The American Institute of Physics, Vol.718, 2004, pp. 497-507.
9. Petukhova N. Development of a Complex Security System in Relational Databases for Railway Transport. In: *Scientific proceedings of Sixth International Baltic Conference on Computer Science and Information Technologies - Databases and Information Systems: DB&IS 2004. Scientific Papers University of Latvia. June 6-9, 2004*. Riga: LU, Vol.673, 2004, pp. 139-150.
10. Петухова Н. Проблемы обеспечения информационной безопасности в темпоральных базах данных, *Transport and Telecommunication*. Vol.7, No3, 2006, с. 504-513.
11. Petukhova N. Investigation of Temporal Data Models in Information Systems on the Railway. In: *Proceedings of the 7th International Conference "Reliability and Statistics in Transportation and Communication" (RelStat'07). 24-27 October 2007. Rīga, Latvija*. Riga: TTI, 2007, pp. 168-177.
12. Копытов Е., Демидов В., Петухова Н. Моделирование расписания в темпоральных базах данных. In: *Proceedings of the International Conference "Modelling of Business, Industrial and Transport System" (MBITS'08). May 7-10, 2008, Riga, Latvia*. Riga: TTI, 2008, pp. 107-116.
13. Копытов Е. А., Демидов В. В., Петухова Н. Ю. Метод построения многоверсионного железнодорожного расписания в информационных системах. В кн.: *Сборник трудов шестой международной научно-практической конференции "Исследование, разработка и применение высоких технологий в промышленности". 16-18 октября 2008 г. Санкт-Петербург, Россия*. Санкт-Петербург: Политехнический университет, с. 17-24.
14. Петухова Н. Темпоральные модели и методы транспортных информационных систем. В кн.: *Материалы международной научной конференции молодых ученых "Наука – будущее Литвы". Транспорт, 3 мая 2007. Вильнюс, Литва*. Вильнюс: ТЕХНИКА (TECHNIKA), 2008, с. 319-325.

15. Kopytov E., Demidovs V., Petukhova N. Application of Temporal Elements in the Railway Schedule Systems. *Transport and Telecommunication*. 2008, Vol.9 No 2, pp. 14-23.
16. Kopytov E., Petukhova N. Application of Temporal Databases in Modeling of Railway Processes. In: *Proceedings of the International Symposium on STOCHASTIC MODELS in RELIABILITY ENGINEERING, LIFE SCIENCE and OPERATIONS MANAGEMENT (SMRLO'10)*. February 8-11, 2010, Beer Sheva, Israel, SCE – Shamoon College of Engineering. Beer Sheva: SCE, 2010, pp. 586-594.
17. Kopytov E., Petukhova N., Demidovs V. Methods for Railway Schedule Periodicity Support in Temporal Databases. In: *Proceeding of 9th JOINT CONFERENCE ON KNOWLEDGE-BASED SOFTWARE ENGINEERING (JCKBSE'10)*. Kaunas, Lithuania, August 25-27, 2010. Kaunas: Tehnologija, 2010, pp. 178-191.

Тезисы

18. Petoukhova N. Increase of safety of data in relational database for records level. В кн.: *Наука и технология - шаг в будущее: Программа и тезисы. Научно-практическая и учебно-методическая конференция, Рига, Латвия, 2-3 мая 2002*. Рига: TSI, 2002, с. 26.
19. Петухова Н. Детализированный доступ к данным, как средство повышения безопасности информационных систем. В кн.: *Proceedings of the International Conference "Reliability and Statistics in Transportation and Communication (RelStat'02)"*, October 17-18, 2002, Riga, Latvia. Рига: ТТИ, 2002, с. 73-74.
20. Kopitovs J., Demidovs V., Petukhova N. Method of Temporal Databases design Using Relational Environment. In: *Proceedings of the 43rd International Scientific Conference of Riga Technical University*. Riga: RTU. 2002. p. 19.
21. Kopitovs J., Demidovs V., Petukhova N. Principles of Creating Data Warehouses in Decision Support Systems of Railway Transport. In: *CASYS '03 – Sixth International Conference on Computing Anticipatory Systems. Liège, Belgium, 11-16 August 2003. Abstract Book*. Liège: CHAOS, 2003, Symposium 8, p. 8.
22. Копытов Е.А., Демидов В.В., Петухова Н.Ю. Применение виртуальных моделей данных в системах принятия решений. В кн.: *Гражданская авиация на современном этапе развития науки, техники и общества. Тезисы докладов Международной научно-технической конференции, посвященной 80-летию гражданской авиации России. 17-18 апреля, 2003*, Москва: МГТУГА, 2003, с. 181-182.
23. Н.Петухова. Принципы разработки комплексных систем безопасности в реляционных базах данных на примере Латвийской железной дороги. В кн.: *Proceedings of the International Conference "Reliability and Statistics in Transportation and Communication (RelStat'03)"*. October 16–17 2003. Riga, Latvia. Рига: ТТИ, 2004, с. 28-29.

24. Petukhova N. Relational Approach to Information Security in Temporal databases. In: *Programme and Abstracts of the 5th International Conference "Reliability and Statistics in Transportation and Communication (RelStat'05)"*. October 13-14 2005. Riga, Latvia. Riga: TTI, 2005, pp. 62-63.
25. Petukhova N. The Problems of Information Security Management in Temporal Databases. *RESEARCH and TECHNOLOGY - STEP into the FUTURE*. Vol.1, No 4. Riga, 2006, pp.115-116.
26. Kopytov E., Demidovs V., Petukhova N. Method of Railway Schedule Supporting in Temporal Databases. In: *Proceedings of the 5-th International Conference "Research, Development and Application of High Technologies in Industry" April 28-30, 2008, St.Petersburg, Russia*. St.Petersburg: Polytechnic University, 2008, pp. 78-79.
27. Kopytov E., Petukhova N. Application of Temporal Databases in Modeling of Railway Processes. In: *Book of Abstracts of the International Symposium on STOCHASTIC MODELS in RELIABILITY ENGINEERING, LIFE SCIENCE and OPERATIONS MANAGEMENT (SMRLO'10)*. February 8-11, 2010, Beer Sheva, Israel. Beer Sheva: SCE, 2010, p. 135

Нормативные документы и концепции

28. Sistēmas „E-rezervēšana starptautiskajos vilcienos” (SAR-R) attīstības koncepcija, Rīga: VAS LDz ISC, 2008, 18 lpp.
29. Sistēmas „E-rezervēšana starptautiskajos vilcienos” (SAR-R) prasību specifikācija, Rīga: VAS LDz ISC, 2009, 43 lpp.
30. Sistēmas „Elektroniskā rezervēšana starptautiskajos vilcienos” (SAR-R) tehnoloģija, Rīga: VAS LDz ISC, 2010, 50 lpp.
31. „Aparatūras un programmatūras nomenklatūras informācijas sistēmas” (APNIS) modernizācijas prasību specifikācija, Rīga: VAS LDz ISC, 2008, 150 lpp.
32. Sistēmas „Interaktīvais vilcienu kustības saraksts” attīstības koncepcija, Rīga: VAS LDz ISC, 2007, 54 lpp.