

А. Н. Базаркин

Основные принципы построения темпоральных реляционных моделей данных

Научный руководитель: к.т.н М. И. Хаткевич

Аннотация. Данная статья представляет собой краткое введение в проблематику теории построения темпоральных реляционных баз данных. В статье представлен небольшой экскурс в историю проблемы, сформулированы определения основных понятий, выполнен краткий обзор основных типов темпоральных баз данных. В статье предлагается общая классификация методов проектирования темпоральных реляционных моделей данных, а также на основе изученного материала формулируются основные принципы построения темпоральных реляционных моделей данных.

1. Введение

В настоящее время на рынке информационных технологий наблюдается уверенная тенденция роста объемов хранилищ данных при снижении их стоимости. Проблема экономии дискового пространства перестает быть актуальной, в связи с чем особый интерес приобретают так называемые темпоральные базы данных (ТБД). Весьма актуальной областью исследований становятся формальные модели данных, ориентированные на хранение темпоральных данных, а также формальные семантики, обеспечивающие взаимосвязь хранимых объектов и атрибутов времени [1].

Традиционные модели данных обеспечивают хранение мгновенного снимка объектов модели предметной области, обычно текущего. Они поддерживают операции изменения, которые приводят к последовательному переходу от одного состояния базы данных (БД) к другому. Любое изменение объекта в базе данных приводит к тому, что предыдущее состояние объекта становится недоступным. В таких моделях предполагается, что запись и извлечение данных — процессы синхронные, то есть информация, хранимая в БД, актуальна на момент выполнения запроса.

В отличие от традиционных систем, темпоральные модели данных (ТМД) позволяют сохранить информацию об эволюции объектов

предметной области: для любого объекта, который был создан в момент времени T_1 и закончил свое существование в момент времени T_2 , в БД будут сохранены все его состояния на временном интервале $[T_1, T_2]$. Таким образом, в ТБД при каждом изменении состояния объекта будет сохраняться новая запись в БД [2].

Под «темпоральностью» следует понимать явную или неявную связь объекта с определенными датами или промежутками времени. В самом широком смысле, темпоральные данные — это данные, которые могут изменяться с течением времени. Под ТМД подразумевается модель данных, ориентированная на хранение темпоральных данных. Таким образом, темпоральная база данных — это БД, построенная на основе темпоральной модели данных и содержащая темпоральные данные.

2. Постановка задачи и обоснование актуальности

Несмотря на то, что многие приложения успешно функционируют на основе традиционных систем управления базами данных (СУБД), существующих возможностей явно недостаточно для информационных систем, динамика изменения информации в которых является одним из ключевых моментов, например, для систем поддержки принятия решений, аналитических систем на основе OLAP технологий. Существует множество прикладных областей, где требуется не только восстановление более ранних состояний базы данных на определенную дату в прошлом, но и создание состояний базы данных на момент времени в будущем (например, задачи планирования). При этом зачастую темпоральные возможности в этих приложениях рассматриваются как некоторые специальные возможности, несмотря на то, что темпоральность должна быть интегрированной частью любой модели данных. Примерами таких типов приложений могут служить финансовые информационные системы (история изменения данных рынка), страховые информационные системы (периоды действия полиса), медицинские информационные системы (записи о пациенте) и системы поддержки принятия решений (прогнозирование будущих показателей). На практике очень трудно найти приложения, которые бы не оперировали темпоральной информацией [3].

В настоящее время на рынке баз данных практически отсутствуют коммерческие СУБД, обладающие полноценными темпоральными возможностями [8]. Для работы с темпоральными данными, как правило, требуется разработка специальных средств и расширение

существующих моделей. Таким образом, вопрос построения ТМД в настоящее время, несомненно, является достаточно актуальным. Теория проектирования ТМД является вполне самостоятельной областью исследований. Однако немногие разработчики приложений осведомлены о том, что многие актуальные проблемы разработки темпоральных приложений изучены, сформулировано множество подходов и методик реализаций различных типов темпоральности [4].

Первые идеи реализации ТМД появились в начале 80-х годов прошлого века в работе Якова Бен-Зви (Jacob Ben-Zvi) в 1982 году [5]. Позже в 80-е гг. начали появляться работы по темпоральной логике и использованию данных, зависимых от времени, их представлению внутри системы и визуализации для пользователя. С тех пор предлагались различные модели, создавались прототипы систем темпоральных баз данных [6]. Одним из ключевых периодов в области исследований темпоральных баз данных, временем ее «официального» представления, можно считать 1992-1993 гг., когда Ричардом Снодграсом (Richard Snodgrass) была предложена идея темпорального расширения стандарта языка запросов к реляционным базам данных SQL-92 [7].

Активными исследованиями в области проектирования ТБД на основе реляционных СУБД занимается интернациональный научный центр TimeCenter. Основной задачей центра является продвижение концептуальных теорий и идей ТМД на практике. В состав центра входит более 30 специалистов в этой области. К настоящему времени существует более 2000 работ, написанных по теме проектирования ТМД за последние 20 лет, однако в отечественной литературе данный вопрос освещен крайне слабо.

При подготовке данной обзорной статьи был изучен ряд работ зарубежных и отечественных авторов, отраженных в списке литературы; выделены основные понятия и методики, на базе которых были сформулированы основные принципы построения темпоральной реляционной модели данных.

3. Основные понятия

В процессе развития теории ТМД было выделено три группы независимых (ортогональных) понятий [9]:

- Типы темпоральных данных.
- Типы времени.

- Темпоральные запросы.

Эти понятия встречаются в любом прикладном приложении, где существует зависимость данных от времени. В свою очередь каждое из этих понятий состоит из трех ортогональных элементов.

Исследователями ТМД выделяется три фундаментальных типа темпоральных данных [9]:

- Момент времени (instant) (событие, которое произошло или произойдет в определенный момент времени, например, сейчас или 1 августа 2009 года в 13.40).
- Интервал времени (interval) (длительность временного отрезка, например, 2 года).
- Период времени (period) (конкретный отрезок времени, например, с 23 апреля 2007 года по 1 августа 2009 года).

Исследователи в области ТМД выделяют три фундаментальных типа времени [9]:

- Время, определяемое пользователем (user defined time) (не интерпретируемое время).
- Модельное время (valid time) (время, в течение которого факт в моделируемой реальности является актуальным).
- Транзакционное время (transaction time) (время физической регистрации факта в БД).

Данные, представленные в БД, можно рассматривать как некоторое отражение текущего состояния действительности в моделируемой реальности, где каждая запись может являться некоторым фактом, являющимся истинным в определенный момент времени. В случае ТБД для каждого факта указывается тот промежуток времени, когда этот факт являлся истинным в моделируемом мире, представленном в БД. Подобное представление времени, когда с данными связывается промежуток времени их актуальности (с точки зрения моделируемого мира), называется модельным или действительным (valid) временем [8].

Другим типом линии времени в теории ТБД является транзакционное время. В любой СУБД каждой записи БД можно сопоставить некоторый промежуток времени, когда данная запись была представлена в БД, то есть промежуток времени между моментами создания и удаления записи в БД [8].

Существует три базовых типа темпоральных запросов [9]:

- Текущие (current) (на текущий момент времени).
- Последовательные (sequenced) (в каждый момент времени).
- Произвольные (non sequenced).

Для каждого момента времени вычисляется реляционный подзапрос, например, «какую зарплату получает каждый из сотрудников?», после чего к общему результату добавляются результаты этих подзапросов с учетом интервалов истинности. Подобная семантика «последовательной» интерпретации реляционных запросов называется последовательной. Однако на основе этой семантики невозможно сформировать запрос, который требует сравнения нескольких последовательных моментов времени. К таким запросам можно отнести большинство запросов, включающих агрегационные функции «во времени», например, «вывести среднюю заработную плату сотрудника за все периоды времени». Поэтому была предложена конструкция запросов произвольного доступа к темпоральным данным, которые предоставляют возможность самостоятельно сформулировать необходимый запрос, накладывая ограничения на системный темпоральный столбец [8]. Как показывает практика, наиболее востребованные запросы — это последовательные, для которых абсолютно отсутствует поддержка в SQL-92. На практике написание запросов для получения последовательных выборок, как правило, является весьма сложной задачей.

Как уже было описано выше, темпоральная база данных — это база данных, построенная на основе ТМД и содержащая темпоральные данные. Однако следует отметить, что не все данные можно назвать темпоральными в полном смысле этого слова, несмотря на то, что они могут быть явным образом связаны с атрибутами времени. Так, добавления атрибута «Дата рождения» к таблице «Сотрудники» не придаст этой таблице свойства темпоральности, поскольку дата рождения является, скорее, неизменяемым во времени атрибутом. Таким образом, предполагается, что темпоральная база данных работает с данными, природа которых изменчива с течением времени. Другими словами, эти базы данных и содержащиеся в них данные могут рассматриваться как темпоральные только в том случае, если известно правило интерпретации временных меток и интервалов для конкретной системы управления базами данных [8].

Согласно определению, понятие ТМД включает в себя три основные компоненты: темпоральная структура данных, темпоральные

ограничения целостности и темпоральные запросы, реализация которых в рамках реляционных моделей данных является отдельной серьезной темой для исследований.

4. Основные типы и особенности темпоральных баз данных

Три типа времени порождают различные типы баз данных [9].

Актуальная БД (Snapshot database) не поддерживает ни действительное, ни транзакционное время. Для этого вида БД не существует возможности просмотра предыдущего состояния после того, как транзакция завершена. Также отсутствует возможность различать периоды актуальности данных в моделируемом мире и время их физического внесения или удаления в БД.

Историчная БД (Historical database) хранит историю изменений сущностей относительно линии модельного времени. Историчные БД требуют наличия более сложного языка запросов, обеспечивающего выборку данных на определенное состояние БД. Для операций изменения требуется явно указывать, к какому из состояний БД необходимо применить изменения.

Транзакционная БД (Rollback database) обеспечивает поддержку транзакционного времени, что делает возможным откат состояния БД на определенный момент времени в прошлом. Этот тип базы данных фиксирует все ошибки и обеспечивает возможность для полноценного аудита данных.

Битемпоральная БД (Bitemporal Databases) хранит как действительное, так и транзакционное время и сочетает в себе свойства обоих типов БД. Этот тип БД позволяет изменять данные в прошлом, так как доступны полные истории изменений и оригинальные значения атрибутов.

5. Классификация принципиальных подходов к реализации темпоральной базы данных

Модифицирование реляционной модели данных для поддержки темпоральных данных означает также, что изменения также производятся и в СУБД. Однако устройство большинства СУБД представляет собой «черный ящик», изменения в котором не представляются возможными. Поэтому основные способы обеспечения поддержки темпоральных данных заключаются в поддержке темпоральной

функциональности на уровне приложения, а также в развитии реляционной модели данных до темпоральной. Фактически, существует четыре различных способа реализации ТБД [10].

5.1. Реализация темпоральной поддержки в приложении

Метод реализации темпоральности на уровне приложения предполагает разработку специальных средств поддержки темпоральности на уровне приложения. Однако на практике данный подход приводит к существенным проблемам, например, когда требуется изменить или заменить часть кода в приложении. Темпоральная семантика в таком случае проектируется каждым разработчиком заново. Темпоральная логика, реализованная на уровне приложения, может быть удобным сиюминутным решением, но не дальновидной стратегией проектирования ИС.

5.2. Использование абстрактных типов для времени

С помощью абстрактных типов данных (АТД) могут быть определены некоторые новые структуры данных, например, набор интервалов времени и операций вычисления, объединения, пересечения в рамках этих интервалов [11]. В конечном счете, АТД могут быть использованы для построения темпоральной семантики в прикладных программах. Одним из основных недостатков этого подхода является тот факт, что СУБД не может использовать специальные семантики времени, например, для оптимизации поиска темпоральных данных. Расширение функциональности СУБД за счет использования АТД приобретает смысл в тех ИС, где дополнительная функциональность касается только части данных либо она очень специфична. Однако на практике темпоральные данные и темпоральные операции обычно не ограничиваются небольшой частью данных и не обладают особой спецификой. Фактически, в большинстве современных приложений достаточно большая часть данных является темпоральными.

5.3. Расширение нетемпоральной модели данных

Расширение нетемпоральной модели данных до темпоральной модели означает, что для спецификации темпоральных понятий используются основные концепции, поддерживаемые нетемпоральной моделью данных. В тех случаях, когда нет возможности выразить некоторые понятия средствами нетемпоральной модели данных, вводятся новые понятия. Язык запросов и алгебра расширяются дополнительными операциями для того, чтобы иметь возможность описывать темпоральные операции объединения или выбора данных. В результате возникает так называемая темпоральная алгебра. Этот подход расширения схемы и языка запросов подходит как для реляционной модели данных ([12]), семантической модели данных, так и для объектной модели данных [10]. На практике этот подход расширения схемы данных наиболее широко используется для определения ТМД. Его преимущество состоит в том, что требуется изменение лишь отдельных частей модели, таких как, например, язык запросов или ограничения целостности. Метод доступа к информации и структура данных остаются без изменений. Из этого преимущества, тем не менее, напрямую вытекает и основной недостаток подхода. Из-за повторного использования существующих понятий нетемпоральной модели данных, ТМД автоматически наследует ограничения нетемпорального родителя.

5.4. Обобщение (генерализация) нетемпоральной модели

Более перспективным способом построения ТМД является обобщение (генерализация) модели данных. Все три компоненты модели (структура данных, операции и ограничения целостности) должны быть обобщены. При неизменной структуре данных типы или схемы объектов не просто расширяются, а вводится новое, простое и ортогональное понятие, которое не зависит от каких-либо типов данных. Проще говоря, это означает, что оно должно быть удобнее в использовании и достаточно выразительным, чтобы помечать различные единицы информации темпоральными метками. Ортогональность в этом контексте означает, что понятие не ограничено особенными конструкциями модели данных, например, кортежами или атрибутами.

Два первых подхода не предполагают каких-либо изменений в СУБД. За основу берется система «как есть», и все наработки и внедрения производятся разработчиком приложения. Два других подхода могут быть реализованы только за счет изменений в СУБД. В настоящее время только два первых подхода могут быть использованы на практике. К сожалению, до сих пор не реализовано полноценной успешной темпоральной СУБД и с трудом может быть разработан прототип такой системы. Предложено множество ТМД, но очень немногие из них внедрены на практике [8].

6. Особенности реализации темпоральной модели данных

Предположим, что имеется таблица, для каждой записи которой необходимо определить даты начала и окончания актуальности. Например, это может быть таблица должностей сотрудников в учреждении — PERSON_APPOINTS (PERSON_ID, APPOINT_ID). В таблицу добавляется два поля FROM_DATE и TO_DATE, которые позволяют получить больше интересных и актуальных запросов, чрезвычайно усложняя ранее безвредные конструкции, такие как первичные и внешние ключи. Добавленные поля в терминах ТБД являются модельным временем, определяющим даты актуальности записи в моделируемом нами мире.

Без атрибутов актуальности записи первичный ключ оригинальной таблицы представлял собой пару (PERSON_ID, APPOINT_ID), однозначно определяющей одну строку в таблице. С добавлением дат актуальности записи определение первичного ключа дополняется условием уникальности в определенный момент времени. В итоге традиционный первичный ключ в SQL перестает быть адекватным для темпоральных таблиц.

Как было отмечено выше, СУБД основаны на модели данных, определяющей конструкции и формализм, с помощью которых все данные могут быть описаны, модифицируемы и извлекаемы универсальным способом. Модель данных поддерживает набор понятий для описания структуры данных, ограничений целостности, выборки и модификации. Как известно, модель данных $M = (DS, OP, C)$ состоит из трех компонент: структура данных DS , операции OP и ограничения целостности C . Темпоральная модель данных $MT =$

(DST, OPT, CT) должна поддерживать все понятия, входящие в каждую из трех компонент, с учетом изменений данных во времени. Алгебра и операции модификации должны быть переопределены, используя темпоральную семантику [10]. Дополнительно, для каждого ограничения целостности в нетемпоральной модели данных M , темпоральная модель данных MT должна поддерживать темпоральную версию. Семантика темпоральных ограничений целостности также должна быть переопределена.

6.1. Модификация данных в темпоральной модели

Как и запросы, в ТБД модификации могут быть трех типов: текущие, последовательные и произвольные [9]. Модификации в ТМД осложняются тем, что могут порождать разрывы или наложения временных интервалов или периодов, что нарушает понятия об ограничениях целостности и первичных ключах. Таким образом, любые модификации должны гарантировать целостность данных, полноту и непротиворечивость.

6.1.1. Текущие модификации в БД

Удаление для текущего состояния БД представляет собой обновление даты актуальности записи и пометку записи как удаленной. При обновлении записи в общем случае должна быть создана новая запись в БД, периодом актуальности которой является промежуток от текущего момента, не ограниченный верхней границей. Также требуется закрытие периода актуальности текущим моментом времени у предыдущей записи и, в общем случае, обновление всех записей, период актуальности которых пересекается с периодом добавляемой записи.

6.1.2. Последовательные модификации в БД

Последовательные модификации предполагают обновление записи в некоторый период в прошлом, то есть границы актуальности этого периода на оси модельного времени располагаются до настоящего момента. В общем случае, перед тем как добавлять запись, необходимо убедиться, удовлетворяются ли условия уникальности и ссылочной целостности для данной записи. Должна быть выполнена проверка на существование записей-дублей для указанного периода,

проверка на наличие ссылок, актуальных с даты начала добавляемого периода, в других таблицах. Записи проверяются на наличие разрыва временной оси в результате планируемых модификаций. В общем случае, для того, чтобы выполнить удаление или обновление записи для состояния БД в прошлом или будущем, необходимо создать новую запись с соответствующими датами актуальности, обновить периоды актуальности записей таким образом, чтобы была обеспечена непрерывность линии времени, а также, если потребуется, удалить записи, который входят в период актуальности удаленной записи.

6.1.3. Произвольные модификации в БД

Произвольные модификации встречаются довольно редко и достаточно сложны в реализации. Например, произвольным обновлением является обновление записи в прошлом — изменение некоторого периода актуальности на некоторый период времени. Произвольные изменения оперируют темпоральными данными произвольным образом, являются достаточно редкими и должны рассматриваться отдельно в каждом конкретном случае [8].

6.2. Темпоральные первичные ключи

Для таблиц с темпоральной поддержкой также требуется модификация состава первичных ключей (ПК) — необходимо включить в них темпоральные атрибуты. Для таблицы PERSON_APPOINTS значение пары атрибутов (PERSON_ID, APPOINT_ID) уникально в любой момент времени. После добавления темпоральных атрибутов в таблице может быть несколько записей с одинаковым значением пары атрибутов (PERSON_ID, APPOINT_ID). Для того, чтобы исключить такую возможность, в набор атрибутов ПК (PERSON_ID, APPOINT_ID, FROM_DATE) может быть включен темпоральный атрибут FROM_DATE. Также в состав ПК могут быть добавлены дата окончания или пара атрибутов (FROM_DATE, TO_DATE), однако ни один из трех вариантов не удовлетворяет требованиям ПК. Проблема возникает в том случае, когда происходит пересечение периодов для одого и того же значения PERSON_ID. Для того, чтобы гарантировать уникальность пары (PERSON_ID, APPOINT_ID) в любой момент времени, необходимо использовать так называемое последовательное ограничение для каждого момента времени. Данный тип темпорального ПК реализуется стандартными средствами SQL

и представляет собой SQL-выражение, осуществляющее проверку на уникальность набора атрибутов [10].

6.3. Уникальность

Условия уникальности в ТМД также имеют особенности. Две записи в случае текущих модификаций называются эквивалентными, если значение их нетемпоральных атрибутов идентичны [9]. Интуитивно это означает, что одна запись в таблице PERSON_APPOINTMENTS, относящаяся к одному работнику, не может быть продублирована позже. В случае последовательных модификаций две записи считаются дублями, если они являются дублями в некоторый момент времени. Интуитивно это следует понимать, как «работник не может иметь несколько одинаковых позиций за один и тот же период». Однако в ТМД это ограничение может быть легко нарушено добавлением очередной записи, период актуальности которой отличается, например, на один день от предыдущей. В случае последовательных модификаций в БД необходимо использовать так называемые последовательные ограничения [10], которые должны обеспечивать уникальность записей с учетом наложений и пересечений периодов актуальности. Эти ограничения реализуются средствами SQL и представляют собой SQL-выражение, осуществляющее проверку.

6.4. Ограничение целостности

Темпоральные ограничения ссылочной целостности в ТМД также имеют некоторые особенности. В общем случае может быть рассмотрено четыре различные ситуации в зависимости от того, для каких таблиц реализована темпоральная поддержка. В случае, если ни одна из таблиц БД не является темпоральной, а также в случае, когда темпоральной таблице является только ссылающаяся таблица, ограничения целостности реализуются традиционными нетемпоральными конструкциями SQL. Иная ситуация возникает в тех случаях, когда таблица, на которую указываются ссылки, является темпоральной, либо все таблицы БД являются темпоральными. Ситуация в

этом случае осложняется тем, что реализация темпоральности может порождать различные наложения периодов и разрывы временной оси. Таким образом, потребуются использование понятия темпоральных ключей и темпоральной уникальности записей. Также следует отметить, что для каждого из типов запросов (текущий, последовательный, произвольный) существует отдельный тип ограничения целостности, имеющий свои особенности [9].

6.5. Темпоральная алгебра и логика

Темпоральная алгебра может быть получена из реляционной алгебры путем расширения ее пяти базовых операций и добавлением отдельных темпоральных операторов. Темпоральная алгебра основана на темпоральной логике [9].

6.6. Нормальные формы

Актуальным направлением в теории ТМД является исследование вопроса соответствия темпоральных моделей нормальным формам. В результате реализации темпоральной поддержки в существующей модели данных встает вопрос наследования свойств при переходе от нетемпоральной модели к темпоральной. Данный вопрос является достаточно важным, поскольку во многом определяет такие свойства как целостность, избыточность и полноту, а также быстрдействие запросов к БД [9].

7. Выводы

Время является одним из главных аспектов, характеризующих многие явления реального мира. Способность моделирования этого темпорального измерения в реальном мире имеет большое значение. Темпоральными базами данных являются базы данных, включающие в себя особую поддержку временного фактора, а также возможности для хранения, извлечения и обновления темпоральных данных. Темпоральная модель данных — это модель данных, ориентированная на хранение темпоральных данных, все аспекты которой также должны быть темпоральными.

Для реализации темпоральных возможностей в рамках ИС программистам, как правило, приходится разрабатывать специальные средства, расширяющие и дополняющие существующие реляционные

модели. Весьма распространенной проблемой разработки таких приложений является отсутствие полного понимания того, каким образом и на каком уровне должна быть осуществлена поддержка темпоральности в БД. Многими разработчиками, реализующими темпоральность в ИС, не учитывается тот факт, что за несколько десятилетий существования данной области исследований накоплено множество различных подходов и методик, изучение которых помогло бы избежать многих традиционных ошибок и заблуждений.

В настоящее время на рынке коммерческих баз данных практически отсутствуют СУБД, обладающие полноценными темпоральными возможностями [8]. Единственным наиболее перспективным решением в сложившейся ситуации может быть построение ТМД в рамках расширения реляционной модели. Понятие ТМД включает в себя темпоральные структуры данных, темпоральные ключи и ограничения целостности, а также темпоральные запросы [9].

В данной работе представлено краткое введение в проблематику темпоральных баз данных, раскрыты основные понятия и подходы к построению ТМД. Исследования в области ТМД являются достаточно перспективными и востребованными. Сообщество по изучению данной проблемы включает в себя несколько сотен исследователей, которыми было написано порядка 2000 работ за последние десятилетия. Некоторые из этих работ представлены в списке используемой литературы и были использованы при подготовке данной обзорной статьи.

Список литературы

- [1] Clifford J. A model for historical databases. — New York University: Center for Research on Information Systems, 1982. ↑1
- [2] Сергеев Г. Методы индексирования исторических данных, Эл. ресурс: <http://www.chair36.msiu.ru/science/science/articles/3/html/node54.html>. ↑1
- [3] Snodgrass R. The TSQL2 Temporal Query Language. — USA: Kluwer Academic Publishers, 1995. ↑2
- [4] Jensen C. S., Soo M. D., Snodgrass R. T. Unifying Temporal Data Models Via a Conceptual Model: Information Systems Vol 19, No. 7, 1994. — 513-547 с. ↑2
- [5] Ben-Zvi J. The Time Relational Model. — PhD thesis. — UCLA: Computer Science Dept., 1982. ↑2
- [6] Bohlen M. H. Temporal Database System Implementations: ACM SIGMOD Record, 1995. ↑2
- [7] Segev, Arie J. S. (Snodgrass, Richard T.) Report on The 1995 International Workshop on Temporal Databases: ACM SIGMOD Record, 1995. ↑2

- [8] Костенко Б. Б. История и актуальные проблемы темпоральных баз данных. — МГУ, 2007, Эл. ресурс: <http://www.citforum.ru/database/articles/temporal>. ↑2, 3, 5.4, 6.1.3, 7
- [9] Snodgrass R. Developing Time-Oriented Database Applications in SQL: Morgan Kaufmann Publishers, 1999. ↑3, 4, 6.1, 6.3, 6.4, 6.5, 6.6, 7
- [10] Steiner A. A Generalisation Approach to Temporal Data Models and their Implementations. — Doctoral Thesis. — Switzerland: Department of Computer Science, 1998. ↑5, 5.3, 6, 6.2, 6.3
- [11] Steiner, A. N. M. C. Implementing Temporal Databases in Object-Oriented Systems: Database Systems for Advanced Applications (DASFAA), 1997. — 381-390 с. ↑5.2
- [12] Sarda N. HSQL: A Historical Query Language: Benjamin/Cummings Publishing Company, 1993. — 110-138 с. ↑5.3

A. N. Bazarkin. *Main principles of temporal relational data models construction* // Proceedings of Junior research and development conference of Ailamazyan Pereslavl university. — Pereslavl, 2009. — p. 23–37. (*in Russian*).

ABSTRACT. In paper short introduction in the temporal relational databases construction theory is represented. In paper small digression to problem history is descibed, definitions of the basic concepts are formulated, the short review of the temporal databases basic types is executed. In paper the general classification of methods of designing temporal relational models of the data is offered, and also on the basis of the studied material main principles of temporal relational data models construction are formulated.