

А. Н. Базаркин

Представление темпоральных данных в МИС Интерин PROMIS

Аннотация. В статье исследуются основные методы представления темпоральных данных в реляционных СУБД. Изучаются проблемы и особенности представления темпоральных данных в МИС. В работе приводятся несколько критериев классификации методов построения темпоральных моделей, формулируются основные подходы к их построению. На пример подсистемы управления персоналом МИС Интерин PROMIS дается подробное описание темпоральной модели данных. На основе полученных результатов формулируются выводы об эффективности различных подходов и методов построения темпоральных моделей данных в МИС.

Ключевые слова и фразы: медицинская информационная система, темпоральные данные, темпоральность.

Введение

Данные в медицинской информационной системе (МИС) связаны с определенными моментами или промежутками времени или, говоря другими словами, являются темпоральными. Информация в интегрированных МИС динамична: в системе, где работают сотни пользователей, обновление информации может происходить каждую секунду. Темпоральные данные, как объект информационных потоков лечебно-диагностического учреждения (ЛПУ), могут иметь принципиальные различия как по типу передаваемой информации (например, медицинские данные, служебные данные), так и по формату (например, графические, текстовые). Динамика темпоральных информационных процессов, и, как следствие, количество типов и форматов темпоральных данных, существенно возрастают в результате взаимодействия ЛПУ с другими учреждениями, а также в результате функционирования ЛПУ в рамках единого информационного пространства (ЕИП). Особое значение приобретает анализ темпоральных данных в точках

соприкосновения различных информационных потоков ЛПУ, учитывающий их темпоральные характеристики и противоречия.

Важным условием четкого функционирования ЛПУ является эффективный сбор и анализ статистических показателей за некоторый период в прошлом, где существенное влияние на качество принимаемых решений может оказывать глубина анализа данных. Процесс принятия решений, касающихся развития медицинского учреждения, либо самого лечебного процесса, должен опираться на достоверные и актуальные темпоральные данные. Однако, в силу того, что на сегодняшний день в целом еще не создано удовлетворительного теоретического, методологического и инструментального базиса для построения интегрированных информационных систем [1], а также ввиду отсутствия полноценных темпоральных системы управления базой данных (СУБД), задача представления темпоральных данных в интегрированной МИС является крайне актуальной и, как правило, становится серьезным препятствием на пути разработки систем такого класса.

1. Темпоральность в МИС

Под «темпоральностью» объекта 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, и очевидно, что под это определение попадают практически все данные МИС.

Для эффективной работы с темпоральными данными требуются адекватная структура хранения темпоральной информации, средства обеспечения ее целостности и непротиворечивости, а также эффективные средства манипуляции данными. Всем этим требованиям, по определению, должны удовлетворять темпоральные СУБД. Под темпоральной СУБД понимается такая СУБД, которая не только обеспечивает хранение темпоральной информации, но и поддерживает некоторые специальные правила хранения и извлечения информации, корректно интерпретирует семантику темпоральных меток.

Как правило, МИС проектируются на основе традиционной реляционной модели данных, к преимуществам которой следует отнести такие качества, как популярность и изученность, удобство использования двумерных таблиц, эффективность доступа к хранимым данным, простота языка манипулирования данными, а также поддержка целостности данных. Однако этой модели присущ один очень важный

недостаток — она «игнорирует» динамику изменения данных во времени. Реляционная модель отражает только текущий «снимок» моделируемой реальности, не предоставляя разработчику эффективных и адекватных инструментов для работы с темпоральными данными.

Проблема представления темпоральных данных в МИС прежде всего обусловлена отсутствием на рынке коммерческих СУБД полноценных темпоральных СУБД [3], а также отсутствием адекватной поддержки работы с темпоральными данными в современных реляционных СУБД. Таким образом, задача проектирования темпоральной модели данных целиком ложится на плечи разработчиков МИС. 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 должного 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.

1.1. Особенности представления темпоральных данных в МИС

- Разнородность темпоральной информации в ЛПУ, что накладывает различные требования на темпоральную модель данных. Модели данных должны учитывать такие факторы, как вид времени для темпоральных объектов, полнота и глубина темпоральной информации.
- Противоречивость требований к представлению темпоральных данных, оказывающихся на стыке различных информационных процессов ЛПУ. Примером таких данных могут служить медицинские услуги, с одной стороны, относящиеся непосредственно к лечебно-диагностическому процессу, а с другой стороны являющиеся предметом финансово-экономической отчетности, что накладывает различные требования в зависимости от подсистемы, оперирующей услугами.

- Наличие в составе интегрированной МИС систем и подсистем различных типов (например, фактографические, документальные). Для каждого из типов существуют различные оптимальные варианты работы с темпоральными данными. Так, например, обработка темпоральных данных в фактографических и документальных системах осуществляется по-разному в силу различных требований к свойствам темпоральной информации.
- Обеспечение взаимодействия различных подсистем, интеграции и синхронизации данных. Достаточно сложной является задача синхронизации и обмена темпоральными данными между подсистемами, хранящих темпоральные данные в различных структурах и по-разному их интерпретирующих.

2. Темпоральная модель данных

Полноценное представление темпоральных данных предполагает полную или частичную реализацию темпоральной модели данных. Темпоральная модель данных (ТМД) — это модель данных, ориентированная на хранение темпоральных данных, все аспекты которой также должны быть темпоральными. Традиционная 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 каждое из трех компонент, с учетом изменений данных во времени [2]. Структура данных должна быть адаптирована таким образом, чтобы она могла хранить темпоральные данные. Алгебра и операции модификации должны быть переопределены, используя темпоральную семантику. Дополнительно, для каждого ограничения целостности в нетемпоральной модели данных M , темпоральная модель данных MT должна поддерживать темпоральный аналог нетемпорального ограничения. Семантика темпоральных ограничений целостности также должна быть переопределена.

Таким образом, разработка ТМД предполагает развитие следующих составляющих:

- Темпоральная структура данных.
- Темпоральные ограничения целостности, ключи.
- Темпоральные запросы и модификации.
- Темпоральная алгебра.

Рассматривая данные, представленные в БД, в качестве некоторого отражения текущего состояния действительности в моделируемом мире, каждая запись может быть воспринята как некоторый факт, который является истинным в определенный момент. При переходе к ТБД для каждого факта можно указать промежуток времени, в течение которого этот факт являлся истинным в моделируемом мире, представленном в БД. Представление времени, когда с данными связывается промежуток времени их актуальности (с точки зрения моделируемого мира), называется модельным или действительным (valid) временем [3].

Другим типом линии времени в теории ТБД является транзакционное время. В любой СУБД каждой записи базы данных можно сопоставить некоторый промежуток времени, когда данная запись была представлена в БД, то есть промежуток времени между моментами создания и удаления записи в БД [3].

Исследователями ТМД выделяются три фундаментальных типа темпоральных данных [2]:

- 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).

Битемпоральная модель данных оперирует как модельным, так и транзакционным временем. Именно битемпоральная модель является наиболее востребованной моделью в большинстве ИС. Именно битемпоральная модель является наиболее полной моделью, обеспечивающей полноценное представление темпоральных данных в МИС.

3. Методы представления темпоральных данных

Модифицирование реляционной модели данных с целью обеспечения поддержки работы с темпоральными данными предполагает изменения модели на уровне 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 либо расширение реляционной модели данных до темпоральной.

Фактически, на практике существует два принципиальных подхода к реализации ТМД:

- Реализация темпоральной поддержки на уровне приложения.
- Расширение нетемпоральной модели данных до темпоральной.

В литературе встречаются и другие способы такие, как генерализация модели данных до темпоральной и использование абстрактных типов, однако на практике использование этих подходов имеет ряд существенных сложностей [4]. Метод реализации темпоральности на уровне приложения предполагает разработку специальных собственных 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.

Расширение нетемпоральной модели данных до темпоральной модели означает, что для спецификации темпоральных понятий используются основные концепции, поддерживаемые нетемпоральной моделью данных. Язык запросов и алгебра расширяются дополнительными операциями для того, чтобы иметь возможность описывать темпоральные операции с данными. На практике этот подход расширения схемы данных наиболее широко используется для построения ТМД. Его преимущество состоит в том, что данный метод предполагает изменение лишь отдельных частей модели, например, языка запросов или ограничения целостности. Метод доступа к информации и структура данных остаются неизменными.

В рамках данного подхода предложены различные ТМД. Принципиальные отличия этих моделей друг от друга можно разделить по следующим критериям:

- Тип темпоральных данных (дискретное или интервальное представление времени).
- Обеспечение темпоральности на уровне отдельных атрибутов или на уровне кортежа.

Темпоральные данные могут быть связаны как с дискретным представлением времени — моментом времени, так с интервальным представлением. Преимущество модели, основанной на дискретном представлении, заключается в ее простоте с точки зрения поддержки стандарта SQL-92 [5]. Однако связь темпоральных объектов с одним атрибутом времени может усложнить и без того непростые темпоральные запросы и операции. В этом плане проще в реализации оказывается модель с интервальным представлением времени. Одним из недостатков этого подхода является отсутствие поддержки понятия «интервала» в стандарте SQL-92. Это понятие может быть смоделировано посредством использования двух моментов времени.

Второй критерий построения темпоральных моделей данных приводит к появлению нескольких различных моделей данных:

- Модель представления темпоральных данных, предложенная Р. Снодграсом (Snodgrass R.) [6].

Будем считать, что битемпоральное отношение R имеет набор атрибутов $A_1, \dots, A_n, T$, где T — битемпоральный атрибут, определенный на множестве битемпоральных элементов. Тогда R можно записать в следующем виде:

$$R = (A_1, \dots, A_n, T_s, T_e, V_s, V_e).$$

Набор дополнительных атрибутов T_s, T_e, V_s, V_e состоит из атомарных темпоральных атрибутов времени, содержащих дату начала и окончания транзакционного и модельного времени. Данная модель данных является самым естественным и наиболее часто используемым способом представления битемпоральных отношений.

- Модель представления темпоральных данных, предложенная К. Дженсеном (Jensen C. S.) [7].

Особенность данного представления заключается в том, что исторические кортежи никогда не обновляются, то есть доступны только для чтения. Таким образом, это представление данных хорошо подходит для основанного на архивах хранения битемпоральных отношений. Этот подход особенно актуален в свете тенденции увеличения объемов носителей информации с одновременным снижением их стоимости. Битемпоральное отношение R с набором атрибутов $A_1, \dots, A_n$ может быть представлено в следующем виде:

$$R = (A_1, \dots, A_n, V_s, V_e, T, Op).$$

Как и в предыдущей схеме представления данных, атрибуты V_s и

V_e хранят даты начала и окончания актуальности факта в моделируемой реальности соответственно. Атрибут T хранит информацию о времени внесения кортежа в журнал изменений. Запросы на создание и удаление кортежей обозначаются в атрибуте Op соответствующими символами — I (Insert) и D (Delete). Модификации данных представляет собой пару запросов: удаления и создание записи, с одинаковым атрибутом времени T .

- Модель представления темпоральных данных, предложенная С. Гадией (Gadia S. K.) [8].

Данный подход предполагает наличие битемпоральных меток у каждого из атрибутов кортежа, что обеспечивает возможность более гибкого моделирования реальности. Пусть битемпоральное отношение R имеет атрибуты $A_1, \dots, A_n, T$, где T — темпоральный атрибут, определенный на множестве битемпоральных элементов. Тогда битемпоральное отношение R может быть представлено в виде отношений, где каждый из атрибутов имеет свою темпоральную метку:

$$R = (([T_s, T_e][V_s, V_e]A_1), \dots, ([T_s, T_e][V_s, V_e]A_n)).$$

Кортеж состоит из n элементов. Каждый элемент представляет собой тройку значений: транзакционное время $[T_s, T_e]$, модельное время $[V_s, V_e]$ и значение атрибута A_i .

- Модель представления темпоральных данных, предложенная Е. МакКензи (McKenzie E.) [9].

В предложенной модели битемпоральное отношение — это последовательность состояний в модельном времени, проиндексированная транзакционным временем. В кортежах с модельным временем атрибуты имеют свои темпоральные метки. Битемпоральное отношение R с набором атрибутов $A_1, \dots, A_n$ представлено в виде отношения, где каждый атрибут помечается временной меткой:

$$R = (VR, T),$$

где VR — это отношение в модельном времени, T — транзакционное время. Схема состояний модельного времени имеет вид:

$$VR = (A_1V_1, \dots, A_nV_n).$$

Здесь $A_1, \dots, A_n$ — это набор атрибутов, V_i — это атрибут модельного времени, связанный с каждым атрибутом A_i и обозначающий время актуальности значения атрибута A_i в моделируемой реальности.

- Модель представления темпоральных данных, предложенная исследователем Дж. Бен-Зви (Bez-Zvi J.) [10].

Пусть битемпоральное отношение R состоит из набора атрибутов $A_1, \dots, A_n, T$, где T — темпоральный атрибут, определенный на множестве битемпоральных элементов. Тогда R может быть представлено в модели Бен-Зви следующим образом:

$$R = (A_1, \dots, A_n, T_{es}, T_{rs}, T_{ee}, T_{re}, T_d).$$

В кортеже значение атрибута T_{es} (effective start) — это время, когда значение атрибута кортежа начинает быть актуальным. Атрибут T_{rs} (record start) хранит информацию о том, когда T_{es} было сохранено в БД. Аналогично, T_{re} хранит информацию о том, когда факт перестает быть актуальным в моделируемой реальности, а T_{ee} — когда T_{re} было зафиксировано в БД. Последний атрибут T_d указывает на время, когда запись была логически удалена из БД.

Кроме этого, темпоральные модели данных могут быть различными по дополнительным критериям таким, как возможность работы с ошибочно введенными данными. Существует ряд методов, предложенных отечественными авторами, суть которых сводится к расширению традиционной модели до темпоральной посредством введения дополнительных таблиц-связей [11]. Авторы данных работ доказывают жизнеспособность и пригодность данных методов на примере разработки прикладного программного обеспечения, однако реализованные авторами темпоральные возможности не являются полноценными с точки зрения определения ТМД.

4. Темпоральная модель данных в МИС Интерин PROMIS

МИС Интерин PROMIS представляет собой информационную и функциональную среду, объединяющую элементы различных классов медицинских информационных систем. МИС обеспечивает комплексную автоматизацию и информационную поддержку всех служб медицинского учреждения. Важное место в МИС занимает подсистема управления кадрами, ее эффективная работа является необходимым условием нормального функционирования учреждения [12]. Подсистема управления кадрами предназначена для автоматизации работы с кадровым составом учреждения, в функции которой входит сквозное ведение штатного расписания, а также оформление и проведение в системы приказов по кадрам на дату как в прошлом,

так и будущем [13]. Реализации темпоральности в данной подсистеме имеет ряд системотехнических сложностей, для решения которых потребовалось принятие научно обоснованных решений архитектурного и методологического характера.

Далее описаны некоторые моменты реализации темпоральной модели данных за счет расширения существующей реляционной модели. Автором сознательно выбраны наиболее важные аспекты реализации темпоральности, которые, по его мнению, заслуживают наибольшего внимания.

4.1. Выбор метода

В качестве основной модели структуры темпоральных данных выбрана модель, предложенная Ричардом Снодграсом (Snodgrass R.) и реализующая темпоральные характеристики на уровне кортежа. Данный метод выбран по причине наибольшей естественности и простой реализации относительно других методов [4].

4.2. Модель данных

Модель данных подсистемы управления кадрами представляет собой набор более чем из тридцати таблиц, двенадцать из которых являются темпоральными. Темпоральную модель данных мы рассмотрим на примере четырех таблиц — K_PERSONS, K_ISPOLS, K_DOLS и K_DOL_DICT. Таблица K_PERSONS содержит тридцать четыре поля, для примера мы будем рассматривать только некоторые из них: PERSON_ID, FAMILY, NAME. Таблица K_ISPOLS состоит из двадцати восьми полей, включая PERSON_ID и DOL_ID. Таблица K_DOLS содержит восемнадцать полей, включая DOL_ID и DOL_DICT_ID. В таблице K_DOL_DICT два поля.

Таким образом, нас интересуют следующие таблицы и атрибуты:

- K_PERSONS (PERSON_ID, FAMILY, NAME);
- K_ISPOLS (PERSON_ID, DOL_ID, SALARY);
- K_DOLS (DOL_ID, DOL_DICT_ID);
- K_DOL_DICT (DOL_DICT_ID, DOL_NAME).

Таблица K_PERSONS содержит информацию о сотрудниках организации. Таблица K_DOLS содержит список должностей организации, K_ISPOLS содержит информацию о том, какую должность занимает сотрудник. K_DOL_DICT является справочником должностей. PERSON_ID — первичный ключ (ПК) таблицы K_PERSONS,


```

AND I2.ACTUAL_FROM < I1.ACTUAL_TO
AND I1.rowid <> I2.rowid )
AND NOT EXISTS (
    SELECT *
    FROM K_ISPOLS I1
    WHERE I1.K_PERSON_ID IS NULL OR
          I1.DOL_ID IS NULL)

```

4.5. Темпоральная уникальность

Для оригинальной таблицы K_ISPOLS условие уникальности может быть записано в виде UNIQUE (PERSON_ID, DOL_ID). Однако с добавлением темпоральной поддержки в таблицы этого условия оказывается недостаточно. Недостаточным является и добавление одного темпорального атрибута или пары атрибутов UNIQUE (PERSON_ID, DOL_ID, ACTUAL_FROM, ACTUAL_TO), поскольку пара тех же значений (PERSON_ID, DOL_ID) может быть добавлена с датами, отличающимися, например, на один день. В этом случае условие уникальности выполняться не будет. Для решения этих проблем были написаны более сложные условия уникальности следующего вида (Листинг 2).

Листинг № 2 «Темпоральное условие уникальности»

```

WHERE NOT EXISTS (SELECT *
    FROM K_ISPOLS AS I1
    WHERE 1 < (SELECT COUNT(PERSON_ID)
    FROM K_ISPOLS AS I2
    WHERE I1.PERSON_ID = I2.PERSON_ID
    AND I1.DOL_ID = I2.DOL_ID
    AND I1.ACTUAL_FROM <= CURRENT_DATE
    AND CURRENT_DATE < I1.ACTUAL_TO
    AND I2.ACTUAL_FROM <= CURRENT_DATE
    AND CURRENT_DATE < I2.ACTUAL_TO))

```

4.6. Темпоральные ограничения целостности

Добавление темпоральной поддержки порождает три возможных случая ограничения ссылочной целостности:

- Ни одна из двух таблиц не темпоральна.

- Одна из двух таблиц темпоральна.
- Обе таблицы темпоральны.

Наиболее сложным случаем реализации условия ограничения целостности данных является третий случай, когда в обеих таблицах реализована темпоральная поддержка (Листинг 3). Листинг № 3 «Темпоральное ограничение целостности»

```
NOT EXISTS ( SELECT *
              FROM K_ISPOLS AS I
              WHERE NOT EXISTS (
                SELECT *
                  FROM K_DOLS AS P
                 WHERE I.DOL_ID = P.DOL_ID
                   AND P.ACTUAL_FROM <= I.ACTUAL_FROM
                   AND I.ACTUAL_FROM < P.ACTUAL_TO)
              OR NOT EXISTS (
                SELECT *
                  FROM K_DOLS AS P
                 WHERE I.DOL_ID = P.DOL_ID
                   AND P.ACTUAL_FROM < I.ACTUAL_TO
                   AND I.ACTUAL_TO <= P.ACTUAL_TO))
```

4.7. Темпоральные запросы

В теории темпоральных баз данных выделяют три фундаментальных типа запросов и модификаций [2]:

- Текущие (current);
- Последовательные (sequenced);
- Произвольные (non sequenced).

Запросы к оригинальным таблицам до добавления темпоральной поддержки соответствуют текущему состоянию моделируемой реальности. Запросы к битемпоральным таблицам приобретают специфику. Рассмотрим более подробно различные типы запросов.

4.7.1. Текущие запросы

Текущий запрос представляет собой запрос значений на некоторый момент времени в прошлом, то есть создание среза истинности

фактов на произвольную дату. Например, для обычного реляционного запроса «какую зарплату сейчас получает каждый из сотрудников?» можно легко сформулировать его темпоральный аналог «какую зарплату получал каждый из сотрудников в указанную дату?» В этом случае результат запроса останется в рамках реляционного представления [3].

4.7.2. Последовательные запросы

Более сложным случаем являются последовательные запросы. Вполне естественным оказывается запрос «когда и какую зарплату получал каждый из сотрудников?» Здесь уже в результатах запроса появляется линия времени. Алгоритм формирования результатов подобных запросов можно упрощенно представить следующим образом: для каждого момента времени вычисляется реляционный подзапрос «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 [3].

Для случая последовательных запросов рассмотрим более подробно особенности операций выборки и связывания.

Последовательная выборка данных не требует особых средств поддержки темпоральности и достаточно просто реализуется.

Более сложной задачей является связывание двух темпоральных таблиц. Например, для вычисления зарплаты и должности для каждого сотрудника мы должны вычислить значение заработной платы для каждого момента времени.

История заработной платы хранится в таблице K_ISPOLS. Прямое связывание двух таблиц K_ISPOLS и K_PERSONS для каждого момента времени неэффективно, потому что значения заработных плат и информация о сотрудниках остаются неизменными в течение длительных периодов. Поэтому эти таблицы следует связывать с использованием их темпоральных периодов. Возможны два различных случая:

- Период в первой из двух связываемых таблиц является более продолжительным, чем период во второй таблице.
- Период во второй из двух связываемых таблиц является более продолжительным, чем период в первой таблице.

Таким образом, выборки темпоральных данных в случае, когда все таблицы имеют темпоральную поддержку, становятся более сложными и состоят, как правило, из нескольких подзапросов, в зависимости от пересечений и наложений периодов темпоральности. Так, например, запрос на выборку истории зарплат всех сотрудников потребует рассмотрения двух общих случаев наложения периодов таблиц в одном запросе.

4.8. Темпоральные модификации

4.8.1. Текущие модификации

Текущие модификации в общем случае предполагают обновление записи и изменение периода актуальности с некоторого момента в прошлом и по настоящее время [2].

Операции создания записи в этом случае требует дополнительных явных условий проверки в блоке WHERE на ограничения ссылочной целостности и уникальности.

В общем случае текущее удаление записи представляет собой обновление даты окончания периодов модельного и транзакционного времени.

Текущее обновление записи в общем случае состоит из следующей последовательности действий:

- Создание новой записи с соответствующими битемпоральными атрибутами.
- Обновление всех записей, дата актуальности которых больше даты начала действия созданной записи.

4.8.2. Последовательные модификации

Последовательные модификации предполагают обновление записи в некоторый период времени в прошлом, то есть границы актуальности этого периода на оси модельного времени располагаются до настоящего момента [2].

Создание новой записи в случае последовательных модификаций осуществляется при выполнении следующих условий:

- в этом периоде актуальности нет дубликатов записи;
- для этого периода есть актуальное значение в таблице, на которую ссылается новая запись;
- нет разрывов во временной оси модельного времени.

При удалении записи возможны четыре различных варианта пересечения периодов актуальности оригинального периода (ОП) и удаляемого периода (УП): УП целиком входит в ОП, УП начинается в рамках ОП, УП заканчивается в рамках ОП и ОП целиком входит в УП. В общем случае удаление записи в случае последовательных модификаций состоит из следующих действий:

- Копируется запись ОП, атрибут ACTUAL_FROM задается равным датой окончания УП.
- Для этой записи атрибут ACTUAL_TO задается равным датой начала УП.
- Для этой записи атрибут ACTUAL_FROM задается равным датой окончания УП.
- Удаляются записи ОП, которые целиком входят в период УП.

В случае обновления записи также возможны четыре различных варианта взаимного расположения двух периодов — оригинального (ОП) и модифицируемого (МП). Обновление записи при последовательных модификациях представляет собой следующий набор операторов:

- Копируется запись ОП, атрибут ACTUAL_TO задается равным датой начала МП.
- Копируется запись ОП, атрибут ACTUAL_FROM задается равным датой окончания МП.
- Обновляются необходимые атрибуты у тех записей, период актуальности которых пересекаются с МП.
- Атрибут ACTUAL_FROM задается равным датой начала МП для тех записей, период актуальности которых пересекает МП.
- Атрибут ACTUAL_TO устанавливается равным дате окончания МП для тех записей, период актуальности которых пересекает МП.

4.9. Произвольные запросы и модификации

Запросы и модификации произвольного типа оперируют темпоральными данными некоторым произвольным образом, являются достаточно редкими и должны рассматриваться отдельно в каждом конкретном случае. Например, к таким запросам можно отнести запросы, требующие «сравнения» нескольких последовательных моментов времени, обычно включающие агрегационные функции «во

времени», например, «вывести среднюю заработную плату сотрудника за все периоды времени» [3]. Запросы данного типа при реализации ТМД в рамках системы управления кадрами МИС Интерин PROMIS не рассматривались.

5. Выводы

До сих пор производителями коммерческих СУБД не предложено ни одной полноценной темпоральной БД. Единственным наиболее перспективным решением в сложившейся ситуации может быть построение ТМД в рамках расширения реляционной модели.

Особенности представления темпоральных данных в МИС обусловлены свойствами интегрированных МИС, среди которых такие, как разнородность информации, распределенность, классификационная полнота. Также распространенная проблема проектирования ТМД связана с тем, что на начальном этапе описания бизнес-процессов и составления ER-моделей разработчиками уделяется недостаточное внимание темпоральной составляющей таблиц БД, не анализируются должным образом виды темпорального времени и цели реализации темпоральной модели данных. Зачастую в процессе разработки не делается различий между временем регистрации факта в БД и временем его актуальности в реальности, предполагая, что все изменения, вносимые в БД, производятся день в день.

На основе представленного в статье анализа наиболее распространенных методов организации темпоральных данных можно сделать некоторые выводы об эффективности и перспективности использования той или иной модели для учета транзакционного и модельного времени темпоральных данных.

Полноценное представление темпоральных данных предполагает полную или частичную реализацию ТМД, включающей в себя темпоральные структуры данных, темпоральные ключи и ограничения целостности, а также темпоральные запросы и темпоральную алгебру. В зависимости от типа информации и ее формата выбирается та или иная модель темпоральных данных в МИС.

Работа с темпоральными свойствами данных на уровне атрибутов, когда каждый из темпоральных атрибутов имеет свои метки даты начала и окончания отрезка транзакционного и модельного времени, является сложнореализуемым подходом в рамках реляционной

СУБД. Это обусловлено тем, что реляционные СУБД не поддерживают таких темпоральных типов данных, как период времени. Реализация темпоральных меток на уровне кортежей является предпочтительным подходом к построению ТМД, поскольку в этом случае темпоральные данные и типы данных адекватно интерпретируются средствами реляционной СУБД.

В данной работе выполнен обзор и анализ основных подходов к представлению темпоральных данных в реляционной СУБД. На основе одного из них реализована и детально описана ТМД в рамках подсистемы управления персоналом МИС Интерин PROMIS. Подсистема управления персоналом является интегрированной частью МИС Интерин PROMIS, где особенно востребованы темпоральные возможности системы.

Список литературы

- [1] Матвеев Г. Н., Хаткевич М. И. Принцип объектно-реляционного дуализма для построения медицинских информационных систем // Материалы IV Научно-практической конференции <«Институт повышения квалификации Федерального управления медико-биологических и экстремальных проблем при Минздраве РФ»>. М., 2000. ↑[1](#)
- [2] Snodgrass R. *Developing Time-Oriented Database Applications in SQL* : Morgan Kaufmann Publishers, 1999. ↑[2](#), [4.7](#), [4.8.1](#), [4.8.2](#)
- [3] Костенко Б. Б. История и актуальные проблемы темпоральных баз данных : МГУ, 2007, Эл. ресурс: <http://www.citforum.ru/database/articles/temporal>. ↑[1](#), [2](#), [4.7.1](#), [4.7.2](#), [4.9](#)
- [4] Jensen C. S., Soo M. D., Snodgrass R. T. *Unifying Temporal Data Models Via a Conceptual Model* // Information Systems, 1994. **19**, no. 7, p. 513–547. ↑[3](#), [4.1](#)
- [5] JCC's SQL Standards Page : JCC Consulting, 2007, <http://jcc.com/SQL.htm>. ↑[3](#)
- [6] Snodgrass R., Ahn I. A *Taxonomy of Time in Databases* // Proceedings of ACM SIGMOD, 1985. ↑[3](#)
- [7] Jensen C. S., Snodgrass R., Soo M. D. *Extending Normal Forms to Temporal Relations* : Technical Report Department of Computer Science, University of Arizona. Tucson, AZ, 1992, no. TR-92-17. ↑[3](#)
- [8] Gadia S. K., Yeung C. S. A *Generalized Model for a Relational Temporal Database* // Proceedings of ACM SIGMOD, 1998. ↑[3](#)
- [9] McKenzie E., Snodgrass R. *Supporting Valid Time in an Historical Relational Algebra: Proofs and Extensions* : Technical Report Department of Computer Science, University of Arizona. Tucson, AZ, 1991, no. TR-91-15. ↑[3](#)
- [10] Ben-Zvi J. 1982. *The Time Relational Model*, PhD thesis, Computer Science Department, UCLA. ↑[3](#)

- [11] Порай Д. С., Соловьев А. В., Корольков Г. В. *Реализация концепции темпоральной базы данных средствами реляционной СУБД* // Документооборот. Концепции и инструментарий ред. Арлазаров В. Л., Емельянов Н. Е. : Едиториал УРСС, 2004, с. 92-109, Тр. Института системного анализа Российской академии наук. ↑³
- [12] Назаренко Г. И., Гулиев Я. И., Ермаков Д. Е. *Медицинские информационные системы: теория и практика* / ред. Назаренко Г. И., Осипов Г. С.. Москва : Наука. Физматлит, 2005. ↑⁴
- [13] Базаркин А. Н., Хаткевич М. И., Хаткевич Ю. И. *Подсистема управления кадрами в интегрированных медицинских информационных системах* // Программные системы: теория и приложения (Переславль-Залесский, 2006) : Тр. междунар. конф., В 2 т./ ИПС РАН ред. Абрамов С. М.. — М. : Наука. Физматлит, 2006. Т. 2, с. 113–124. ↑⁴

A. N. Bazarkin. *Temporal data representation in the MIS Interin PROMIS.*

ABSTRACT. Basic methods of temporal data representation in relational DBMS are investigated in the article. Problems and features of temporal data representation in MIS are studied. In paper some criteria of methods classification are resulted, the basic construction approaches are formulated. On an example of the personnel registration subsystem of MIS Interin PROMIS the detailed description of the temporal data model is described. On the basis of the received results conclusions about efficiency of various approaches and construction methods temporal data models in MIS are formulated.

Key Words and Phrases: medical information systems, temporal data, temporal data model.

Образец ссылки на статью:

А. Н. Базаркин. *Представление темпоральных данных в МИС Интерин PROMIS* // Программные системы: теория и приложения : электрон. научн. журн. 2010. № 4(4), с. 3–21. URL: http://psta.psiras.ru/read/psta2010_4_3-21.pdf