

Представление темпоральной модели данных в банковской информационной системе

Н. О. Пономарев , email: fores863@gmail.com

Санкт-Петербургский государственный экономический университет

Аннотация. *В статье исследуется темпоральная модель данных для реляционных СУБД. Делается акцент на расширенных функциональных возможностях темпоральной структуры по сравнению с обычными моделями данных. На примере банковской информационной системы описываются особенности структуры таблиц и написание запросов на выборку данных. На основе проделанной работы автор обозначает ключевые векторы развития в данной научной области.*

Ключевые слова: *темпоральная модель данных, темпоральная база данных, структура данных, SQL, СУБД.*

Введение

Время – важный аспект всех явлений реального мира. События происходят в определенные моменты времени; объекты и отношения между объектами существуют с течением времени. Способность моделировать это временное измерение реального мира необходима для многих областей, таких как эконометрика, банковское дело, контроль запасов, бухгалтерский учет, юриспруденция, медицинские записи, наземные и географические информационные системы и бронирование авиабилетов.

Обычные базы данных представляют состояние предприятия в отдельный момент времени. Хотя содержимое базы данных продолжает меняться при добавлении новой информации, эти изменения рассматриваются как модификации состояния, при этом старые, устаревшие данные удаляются из базы данных. Текущее содержимое базы данных можно рассматривать как данные, актуальные на определенный момент времени. В таких системах атрибутами, включающими время, управляют исключительно прикладные программы; система управления базами данных (СУБД) интерпретирует даты как значения в базовых типах данных.

Данные банковской информационной системы существуют в динамике. Внутри банковской системы протекает много изменений: меняются условия договора с клиентами, заключаются дополнительные соглашения, счета клиентов открываются и закрываются. При этом для

построения внутренней отчетности и анализа необходима историчность данных.

В этой статье предпринята попытка охватить и обобщить основные концепции, подходы и стратегии внедрения темпоральной модели данных, которые были сформированы в результате этого исследования. Сначала будет исследована временная область: ее структура, размерность. Далее будет проведен анализ темпоральной модели данных в банковской сфере.

1. Темпоральность в банковской системе

Итак, темпоральность данных предполагает, что в СУБД происходит не хранение актуальной копии данных на текущий момент времени, а данных за все время существования СУБД с темпоральной моделью. С помощью данного подхода появляется функциональность:

- Аудит по изменению данных и экспертиза данных
- Восстановление данных на момент времени
- Вычисление тренда с течением времени
- Поддержание *slowly changing dimensions*
- Восстановление после случайных ошибок в данных и ошибок в загрузчиках

Для внедрения ТМД требуются средства обеспечения ее непротиворечивости и целостности. Под этим понимается не только необходимая структура хранения данных и наличие расширенного списка полей, но и так же специальное обновление, вставка и извлечение данных.

2. Описание темпоральной модели и возможной структуры

Для описания темпоральной модели и возможной структуры приведем примеры двух таблиц в реляционной СУБД.

Первая таблица – «Договоры в группе риска».

Таблица 1

Договоры в группе риска

№ п/п	Поле	Наименование поля	Тип данных	Доп. тип данных
1	agreement_id	ID договора	decimal	Числовой
2	reserve_rate	Ставка резервирования	decimal	Числовой
3	risk_group_num	Номер группы кредитного риска	varchar	Текстовый
4	loan_quality_name	Качество	varchar	Тестовый

		обслуживания долга		
5	effective_from_dt	Дата начала действия записи	date	Дата (Интервальная)
6	effective_to_dt	Дата окончания действия записи	date	Дата (Интервальная)
7	is_effective	Флаг активной записи	tinyint	Числовой
8	deleted_flg	Флаг удаленной записи	tinyint	Числовой
9	last_load_dttm	Дата загрузки записи	date	Дата
10	initial_load_id	Идентификатор запуска процесса загрузки	int	Числовой
11	last_load_id	Идентификатор загрузки записи	int	Числовой

В таблице присутствуют системные поля, предназначенные для создания новой структуры взаимодействия с данными. Опишем ключевые для темпоральной модели. Effective_from_dt и effective_to_dt образуют полуоткрытый интервал [effective_from_dt, effective_to_dt). В свою очередь у каждой записи появляется диапазон, в котором она является валидной. Актуальная же запись имеет открытый интервал в плюс бесконечность. При загрузке актуальной информации о договоре старый интервал закрывается текущей датой, и создается новый интервал.

Рассмотрим таблицу «Договор».

Таблица 2

Договор

№ п/п	Поле	Наименован ие поля	Тип данны х	Доп. тип данных
1	agreement_id	ID договора	decimal	Числовой
2	Agreement_sid	Внутренний	varchar	Текстовый

		ID договора		
3	Agreement_class_id	ID класса договора	decimal	Числовой
4	effective_from_dt	Дата начала действия записи	date	Дата (Интервальная)
5	effective_to_dt	Дата окончания действия записи	date	Дата (Интервальная)
6	is_effective	Флаг активной записи	tinyint	Числовой
7	deleted_flg	Флаг удаленной записи	tinyint	Числовой
8	last_load_dttm	Дата загрузки записи	date	Дата
9	initial_load_id	Идентификатор запуска процесса загрузки	int	Числовой
10	last_load_id	Идентификатор загрузки записи	int	Числовой

В таблице «Договор» структура системных полей аналогична таблице «Договоры в группе риска». Это является не случайным, при создании темпоральной модели данных необходимо, чтобы все объекты в системе имели одну и ту же структуру для правильного понимания данных, а также для правильного анализа данных. Темпоральная модель данных позволяет легко менять актуальную информацию и в таблицах-словарях при необходимости без сложностей с изменением текущей структуры, что позволяет гибко расширять и дополнять модель данных.

3. Темпоральные запросы

Темпоральная модель данных представляет широкий круг запросов на выборку данных. Основные из них рассматриваются в таблице в зависимости от отсекающих условий по темпоральным полям.

Правила на запросы на выборку данных

Выражение на выборку	на	Условие на поля	Примечание
На заданную дату(date_time)		effective_from_dt <= date_time AND effective_to_dt > date_time	Возвращает таблицу со значениями, действительными на указанный момент времени в прошлом.
Интервал (start_date, end_date)		effective_from_dt < end_date AND effective_to_dt > start_date	Возвращает таблицу со значениями, которые были действительны в заданном интервале.
Между start_date и end_date		effective_from_dt <= end_date AND effective_to_dt > start_date	То же, что и в предыдущем примере, но возвращаемая таблица включает строки, которые попадают верхнюю границу
Содержится в [start_date, end_date]		effective_from_dt >= start_date AND effective_to_dt <= end_date	Возвращает все строки, которые были открыты и закрыты в данном отрезке
Вся таблица		- (Нет условия)	Возвращает таблицу с полной историзацией

В данной статье будет рассмотрен пример темпорального запроса на заданную дату N, так как данный тип запросов чаще всего встречается в банковской информационной системе для построения отчетности.

Листинг

Запрос на выборку данных на дату 06.12.2022

```
with dateT AS (
  select '2022-12-06' as date_dok
), example as (
SELECT a.agreement_sid, a.agreement_class_id, arg.*
  from dateT

  inner join dds.t_dim_agreement a on
    a.effective_from_dt <= dateT.date_dok
```

```

and
a.effective_to_dt > dateT.date_dok

inner join dds.t_dim_agreement_risk_group arg on
arg.agreement_id = arg.agreement_id
and
arg.effective_to_dt > dateT.date_dok)
SELECT * FROM example;

```

Запрос был написан с помощью Transact-SQL. В текущем примере была выбрана дата 6 декабря 2022 года. С помощью обобщенного табличного выражения было реализовано слияние двух темпоральных таблиц на заданную дату. По такой же структуре представляется возможным построение более сложных запросов:

- Объединение большего количества таблиц
- Вычисление промежуточных результатов и их использование в качестве части обобщенного табличного выражения
- Применение оконных функций

Заключение

Представление темпоральных данных необходимо во многих ИС, включая банковскую, но до сих пор нет удобного и доступного инструмента для СУБД по хранению таких данных. Это приводит к тому, что большинство темпоральных моделей строятся поверх реляционных баз данных, что требует компетенций для разработчиков на всех уровнях: от проектировщиков модели данных, до конечных потребителей (аналитиков данных).

В данной статье была рассмотрена и описана темпоральная модель данных внутри банковской информационной системы. Также была представлена структура данных, описанная с помощью двух таблиц. На основе представленных таблиц реализован запрос на извлечение данных.

Исходя из динамики роста количества данных и снижения цен на носители информации, в качестве дальнейшего вектора исследования в области темпоральной модели может служить разработка новых алгоритмов хранения и обработки информации, что позволит сделать новый шаг к анализу исторических данных.

Список литературы

1. Костенко Б.Б. История и актуальные проблемы темпоральных баз данных [Электронный ресурс] / Б.Б. Костенко. – Режим доступа: www.citforum.ru/database/articles/temporal

2. А. Н. Базаркин, “Представление темпоральных данных в МИС Интерин PROMIS”, Программные системы: теория и приложения, 1:4 (2010), 3–21

3. Документация Microsoft Temporal Tables : сайт. – Режим доступа: <https://learn.microsoft.com/en-us/sql/relational-databases/tables/temporal-tables?view=sql-server-ver16>

4. Бен-Ган Ицик Оконные функции в T-SQL / Ицик Бен-Ган. – Москва : ДМК-Пресс, 2022 г. – 344 с. – ISBN 978-5-93700-139-9

5. Snodgrass R.T. Adding Valid Time to SQL / Temporal / R.T. Snodgrass. – San Francisco: Morgan Kaufmann Publishers, 2000. – 87 p.

6. Snodgrass R. et al. Temporal databases //Computer. – 1986. – Т. 19. – №. 09. – С. 35-42.

7. Tansel A. U. et al. (ed.). Temporal databases: theory, design, and implementation. – Benjamin-Cummings Publishing Co., Inc., 1993.