УДК 004.652.42

ХРАНЕНИЕ, АНАЛИЗ И ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ В РЕЛЯЦИОННЫХ СУБД

Мордвинов И. В.

Научный руководитель – д.ф.-м. н., профессор Добронец Б. С.

Сибирский федеральный университет

Подавляющее большинство потребителей информационных технологий в той или иной степени уже решили задачу оперативного ввода данных. Однако при этом многие из них испытывают насущную потребность в организации анализа накопленных данных.

Одним из наиболее часто встречающихся объектов накопленных данных являются временные ряды. Временной ряд представляет собой результат наблюдения за одним или несколькими параметрами какого-либо процесса. При наблюдении, значения фиксируются и привязываются к моменту наблюдения. Результатом является упорядоченная в хронологическом порядке последовательность данных, которая и называется временным рядом.

В общем случае, пусть наблюдаемым временным рядом является $y_1, y_2, \ldots, y_n$. Мы будем понимать эту запись следующим образом. Имеется T чисел, представляющих собой наблюдение некоторой переменной в T равноотстоящих моментов времени. Эти моменты для удобства пронумерованы целыми числами 1, 2, …, T. Достаточно общей математической (статистической или вероятностной) моделью служит модель вида:

$$y_t = f(t) + u_t, = 1,2,..., \qquad (1)$$

Модель временного ряда представляет собой уравнение (1), которое связывает наблюдение, полученное в некоторый конкретный момент времени, с наблюдениями, полученными ранее по той же и/или другим характеристикам изучаемой переменной.

За последние несколько десятилетий разработан обширный математический аппарат анализа временных рядов. Главные задачи этого аппарата — выявлять закономерности в поведении наблюдаемого процесса и прогнозировать его поведение в будущем.

Несмотря на глубокую теоретическую проработку методов анализа временных рядов, их практическая реализация в настоящее время встречает серьезные трудности. В числе наиболее серьезных источников проблем следует назвать неспособность крупных промышленных реляционных СУБД (РСУБД) эффективно хранить и обрабатывать временные ряды. Дело в том, что именно этот класс СУБД составляет основу крупных корпоративных информационных систем (КИС). При этом они имеют все необходимые средства для описания временного ряда как одной из хранимых структур данных. Однако производительность обработки этой структуры данных

оказывается столь низкой, а трудоемкость описания прикладной логики столь высокой, что на практике эти СУБД для хранения и обработки временных рядов не используются.

При первом рассмотрении не очень ясно, почему РСУБД не могут справиться со столь простой структурой данных, как временной ряд. Ни для кого не секрет, что РСУБД хорошо зарекомендовали себя как основа для построения крупных информационных систем; они с успехом применяются как для задач оперативной обработки информации, так и в системах поддержки принятия решений. Развитая математическая теория, которая лежит в основе логической модели базы данных, хорошо отработанная технология и наличие поддерживаемых промышленных стандартов способствовали широкому распространению РСУБД.

Главная причина неэффективной работы РСУБД с временными рядами заключается в плохой совместимости основ реляционной модели и природы временного ряда. Одной из фундаментальных основ реляционной модели является понятие отношения — неупорядоченного множества кортежей. Напротив, принципиальным свойством временного ряда является упорядоченность его элементов.

Некоторые производители РСУБД вводят в свои продукты средства оптимизации работы с упорядоченными последовательностями. Так, широкое распространение получили кластерные индексы. При построении такого индекса записи на диске упорядочиваются в соответствии со значениями ключей индекса. Указанный подход позволяет оптимизировать доступ к данным, но при этом проблемы применения реляционной модели к временным рядам полностью не решаются.

В качестве примера временного ряда рассмотрим продажу иностранной валюты в крупном банке. Объектами анализа в таком примере являются код валюты и объем проданной валюты.

Таблица 1. Классический пример временного ряда

Время	Код валюты	Объем валюты	Цена
…	…	…	…
13.03.11 16:43:12	840 (USD)	43	28.91
13.03.11 16:43:13	978 (EUR)	876	41.12
13.03.11 16:43:15	840 (USD)	20	28.91
13.03.11 16:43:15	826 (GBP)	5	51.07
13.03.11 16:43:21	392 (JPY)	84	40.92
…	…	…	…

Записи появляются в хронологическом порядке, по мере совершения сделок. Однако для анализа временного ряда используются данные, относящиеся только к одному объекту, и значительно реже производятся манипуляции с несколькими рядами одновременно. Таким образом, из таблицы будут извлекаться записи, относящиеся к одной валюте, в порядке возрастания или убывания отметок времени сделки.

Проанализируем эффективность доступа к данным в описанной таблице. Длина строки невелика — несколько десятков байт, поэтому в одной странице данных СУБД может помещаться около сотни записей. В большинстве приложений временных рядов одной структуры также около ста. Получается, что при очень грубой оценке для каждого временного ряда записи распределены примерно по одной на каждую страницу дискового пространства. Значит, чтобы выбрать записи, относящиеся к одному временному ряду, необходимо обращаться практически к каждой странице данных. Очень часто на практике размер таблицы значительно превосходит размер буферов СУБД. В этой ситуации почти каждое обращение к элементу временного ряда потребует подкачки новой страницы данных с дисковых накопителей. В таком режиме эффективность индексированного доступа практически равна нулю, поскольку основное время занимают операции ввода-вывода с дисковой подсистемы.

Ситуация может улучшиться в том случае, если необходимо извлечь не весь временной ряд, а только элементы, находящиеся в определенном временном интервале. Тогда индекс помогает ограничить количество извлекаемых страниц данных.

Необходимо обратить внимание на то, что при использовании индексного доступа к элементам временного ряда по аналогичным причинам происходит сканирование значительной части индексного дерева. Здесь удается избежать прохода по всему индексному дереву за счет построения сцепленного индекса по коду ценной бумаги и времени сделки. В таком сцепленном индексе ключи, относящиеся к одному временному ряду, будут локализованы в одном месте дерева. Это позволит последовательно пройти по ключам индекса, относящимся к запрошенному пользователем интервалу времени.

В начале каждого ключа содержится код ценной бумаги, который увеличивает размер индекса, замедляет поиск и (особенно сильно) вставку новых записей. Поэтому индекс, хоть и может помочь при выборке элементов ряда, попадающих в один временной интервал, в целом работает малоэффективно.

Очевидна необходимость выделения записей, относящихся к одному временному ряду, в отдельную область хранения. При использовании классической реляционной структуры это можно сделать только отведя под каждый временной ряд отдельную таблицу. Но тогда для обращения к конкретному временному ряду придется задавать имя таблицы как параметр, то есть придется использовать динамический SQL. Далеко на все инструментальные средства в принципе поддерживают возможность работы с динамическим SQL (например, SQLJ level 0). Кроме того, динамический SQL значительно затрудняет разработку и замедляет выполнение программ. К тому же большое количество таблиц, соответствующее числу временных рядов в базе данных, сильно усложняет администрирование СУБД.

Для хранения временных рядов большого размера на дисковом пространстве сервера выделяется раздел, называемый контейнером. При создании в таблице колонки типа TimeSeries устанавливается размер временного ряда, который может храниться

вместе с другими полями таблицы. Обычно этот размер составляет несколько килобайт. Если временной ряд превышает этот размер, то будет храниться в контейнере.

По своей структуре контейнер мало чем отличается от пространства, выделяемого для хранения обычной таблицы. При его создании описывается структура элементов временного ряда, которые будут в нем храниться. Временные ряды, принадлежащие одной таблице, могут располагаться в различных контейнерах. Одновременно с этим в самом контейнере могут располагаться временные ряды, принадлежащие различным таблицам.

В контейнере для каждого временного ряда по возможности отводится непрерывная область дискового пространства. Внутри этой области элементы хранятся упорядоченно, в хронологической последовательности. Порядок поддерживается даже при изменении отметки времени.

Для повышения эффективности при создании временного ряда можно указать, является ли он регулярным или нерегулярным. Также при создании контейнера устанавливается, какой из этих типов рядов будет храниться в контейнере. У регулярного ряда вместо отметки времени сохраняется величина его смещения от начала, которая кодируется четырехбайтовым числом (для сравнения: полная отметка времени кодируется восьмибайтовым числом). В случае регулярного ряда можно быстро вычислить точное местоположение записи по величине смещения. За счет этого существенно повышается скорость работы с регулярными временными рядами, поскольку для позиционирования в нерегулярном временном ряде используется индекс по отметкам времени.

Развитие реляционной технологии предоставило РСУБД возможность проникнуть в очередную, прежде закрытую для них сферу — обработку временных рядов. Это не только позволяет достичь возможностей, доступных ранее только специализированным системам, но и открывает новые перспективы для данной прикладной области.