УДК 004.65
doi:10.18720/SPBPU/2/id24-201

Морозов Сергей Владимирович [1],
студент бакалавриата;
Нестеров Сергей Александрович [2],
доцент, канд. техн. наук, доцент

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ТИПОВ ИНДЕКСОВ В СУБД SQL SERVER И POSTGRESQL

[1, 2] Россия, Санкт-Петербург,
Санкт-Петербургский политехнический университет Петра Великого;
[1] morozov.sv@edu.spbstu.ru, [2] nesterov@spbstu.ru

Аннотация. Работа посвящена сравнительному анализу индексов для оптимизации запросов в двух популярных СУБД: Microsoft SQL Server и PostgreSQL. Проведен краткий обзор классических и специальных индексов для обеих СУБД. Выполнено сравнение типов индексов СУБД по критерию применимости к различным типам данных.

Ключевые слова: базы данных, оптимизация запросов, индексы, PostgreSQL, SQL Server.

Sergey V. Morozov [1],
Student;
Sergey A. Nesterov [2],
Candidate of Technical Sciences (PhD), Associate Professor

COMPARATIVE ANALYSIS OF INDEX TYPES IN SQL SERVER AND POSTGRESQL DBMS

[1, 2] Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russia;
[1] morozov.sv@edu.spbstu.ru, [2] nesterov@spbstu.ru

Abstract. The work is devoted to a comparative analysis of indexes for query optimization of two popular DBMS: Microsoft SQL Server and PostgreSQL. A brief review of classic and special indexes for both DBMS is provided. The types of DBMS indexes were compared according to the criterion of applicability to various types of data.

Keywords: databases, query optimization, indexes, PostgreSQL, MS SQL Server.

Введение

Выполнение запросов к базам данных может привести к обработке большого количества записей, что может потребовать достаточно много времени. Одним из способов снижения стоимости выборки данных из таблиц является отказ от последовательного сканирования и применение методов прямого доступа, предусматривающих использование специальных указателей (адресных ссылок) на строки таблиц, соответствующие

критерию выборки. Такие методы требуют наличия служебных структур данных, называемых индексами [1, 2].

1. Постановка задачи

Для проведения сравнительного анализа необходимо провести краткий обзор типов индексов для каждой СУБД, разобрать каждый тип, описать его характеристики, структуру и область использования. Это позволит оценить возможность миграции с одной СУБД на другую или выбрать лучшую СУБД для старта проекта.

2. Обзор типов индексов в СУБД SQL Server

СУБД SQL Server поддерживает такие типы индексов, как:

– Кластеризованный индекс;
– Некластеризованный индекс;
– Пространственный индекс;
– XML-индексы;
– Полнотекстовый индекс.

Некластеризованный индекс в SQL Server представляет собой многоуровневый иерархический индекс. На корневом уровне дерева находится единственная индексная страница, а количество страниц на каждом дочернем уровне зависит от количества значений ключа индекса на всех страницах родительского уровня и, в частности, от степени заполнения индексных страниц. На нижнем (листовом) уровне каждая индексная строка содержит указатель на соответствующую строку таблицы базы данных, включающий номер файловой страницы из кучи и номер строки в рамках этой страницы. Каждый уровень дерева является двусвязным списком из индексных страниц, образуя b-дерево (b-tree). Структура подобного индекса схематично представлена на рисунке 1.

Некластеризованные индексы существенно ускоряют поиск данных по уникальным ключам, но при высокой степени селективности предиката выборки их эффективность резко падает. Такие индексы хорошо работают на выборке с точечным условием и хуже, если выборка происходит по диапазону.

Кластеризованный индекс — это структура данных, объединяющая в единый «кластер» и строки индексируемой таблицы, и собственно индекс. Такой индекс имеет структуру, как у некластеризованного индекса, только на нижнем уровне будут находиться страницы с данными таблицы, упорядоченные по значению ключа кластеризованного индекса [1]. Так как данные в индексе хранятся уже в упорядоченном виде, это значительно ускоряет выборку по диапазону значений, а также при реализации методов, требующих сортировки данных.

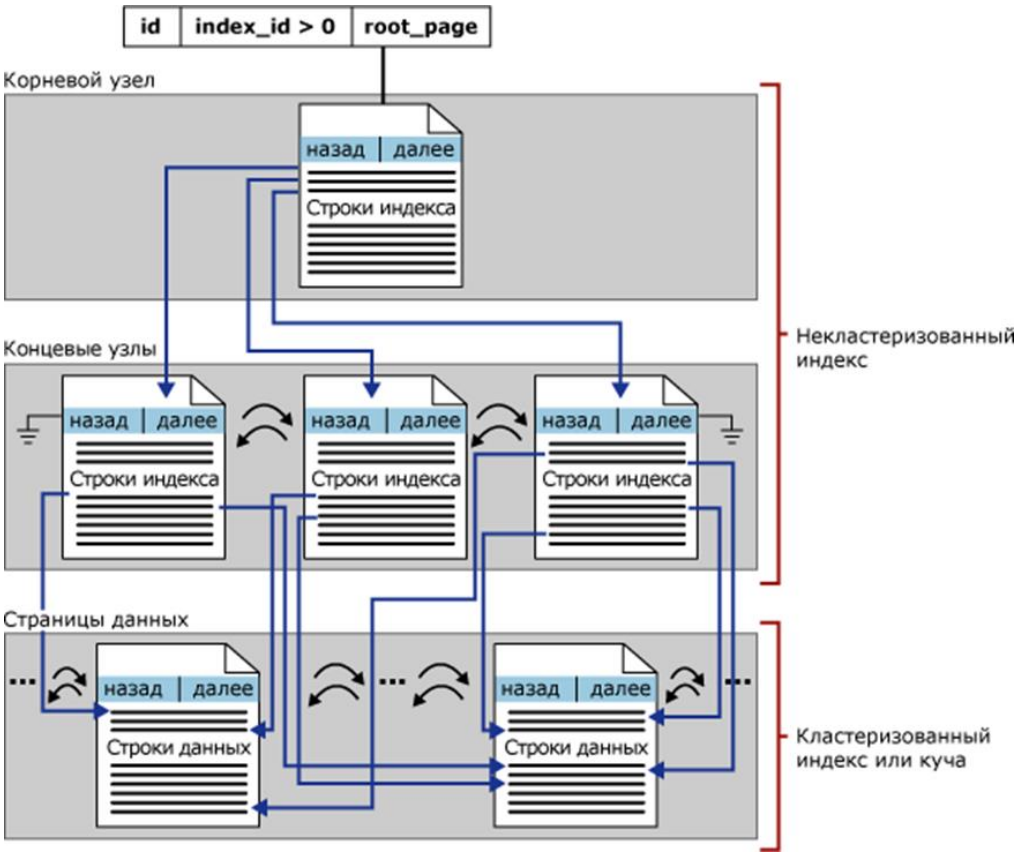

Рис. 1. Структура некластеризованного индекса в SQL Server

В SQL Server *пространственные индексы* создаются с помощью b-деревьев, что означает, что индексы должны представлять 2-мерные пространственные данные в линейном порядке b-деревьев. Таким образом, перед чтением данных в пространственный индекс SQL Server реализует иерархическую однородную декомпозицию пространства. В процессе создания индекса происходит декомпозиция пространства в четырехуровневую сеточную иерархию.

Каждый последующий уровень содержит дальнейшую декомпозицию уровня выше, так что каждая ячейка уровня выше содержит полную сетку следующего уровня. На заданном уровне все сетки имеют одинаковое число ячеек на обеих осях, и все ячейки имеют одинаковый размер.

XML-индексы в СУБД MS SQL Server используются для столбцов типа XML. При этом индексируются все теги, значения и пути хранимых в столбце экземпляров XML и повышается эффективность обработки запросов [4].

XML-индексы разделяются на 2 типа:

1. Первичный XML-индекс.
2. Вторичный XML-индекс.

Первичный XML-индекс — это разобранное и сохраненное представление XML-объектов BLOB (от англ. Binary Large Object — большой двоичный объект), содержащихся в столбце типа данных XML. Для каждого большого двоичного объекта столбца типа данных XML в индексе создается несколько строк данных, и их количество приблизительно равно числу узлов в большом двоичном объекте XML [4].

Вторичный XML-индекс возможен только при наличии первичного XML-индекса. Вторичные XML-индексы бывают трех видов:

– индекс Path, оптимален, если запрос использует выражение пути;
– индекс Value, оптимален, если запрос не знает имён атрибутов в XML;
– индекс Property, оптимален, если известен первичный ключ объекта и используется метод value().

Полнотекстовый индекс предоставляет индексирование, поддерживающее полнотекстовые запросы. В отличие от стандартных индексов, полнотекстовые разбивают данные в столбце на токены, с помощью этих токенов и происходит построение индекса. Вместо создания сбалансированного дерева на основе значения, хранящегося в конкретной строке, служба полнотекстового поиска создает инвертированную стековую сжатую структуру индекса на основе отдельных токенов индексируемого текста [4].

Данный специальный тип индекса используется при создании индекса на столбцах, не индексирующихся индексами стандартных типов. Проблемой использования этого индекса является большой объём дискового пространства, занимаемый таким индексом. Эта проблема частично решается механизмом «стоп-слов» (часто встречающихся слов, удаляемых из индексируемого текстового объекта) и сжатием хранимых в индексе данных.

3. Обзор типов индексов в СУБД PostgreSQL

СУБД PostgreSQL поддерживает такие типы индексов, как:

– B-tree индексы;
– GiST индексы;
– SP-GiSTиндексы;
– GIN индексы;
– BRIN индекс.

B-tree индексы устроены так же, как и многоуровневый иерархический индекс, описанный ранее (рис. 1). Листовой уровень в b-tree в PostgreSQL упорядочен по аналогии с кластеризованным индексом в SQL Server. Имеет те же преимущества и недостатки (проблема перестроения при модификации). Он используется в большинстве случаев.

GiST индекс (от англ. generalized search tree, обобщенное поисковое дерево) имеет структуру сбалансированного дерева поиска, так же, как и

b-tree. GiST индекс позволяет задать принцип распределения данных произвольного типа по сбалансированному дереву, и метод использования этого представления для доступа по некоторому оператору.

Каждая запись листового узла содержит некий предикат и ссылку на строку индексируемой таблицы. Узлы дочернего уровня так же содержат предикат и ссылку на дочерний узел, причём все индексированные данные дочернего поддерева должны удовлетворять этому предикату.

Для поиска по дереву GiST индекс использует функцию согласованности — одну из функций, определяемых интерфейсом, и реализуемую по-своему для каждого поддерживаемого семейства операторов. Эта функция вызывается для индексной записи и определяет согласованность предиката с условием поиска.

Чаще всего GiST индексы используются в задачах, связанных с географическими или геометрическими объектами и полнотекстовом поиске.

SP-GiST индекс (от англ. Space-Partitioned GiST, GiST с разбиением пространства) поддерживает деревья поиска с разбиением, что облегчает разработку широкого спектра различных несбалансированных структур данных. SP-GiST индекс разбивает область значений на неперекрывающиеся подобласти, которые тоже могут быть разбиты. Неперекрываемость подобластей существенно ускоряет задачи вставки и поиска.

Устройство таких индексов аналогично GiST индексам, только для каждой ссылки во внутреннем узле может быть задана метка. Листовой узел и алгоритм поиска аналогичен GiST индексам.

GIN индекс (от англ. Generalized Inverted Index, обобщённый инвертированный индекс) работает с неатомарными данными, индексируя при этом отдельные элементы (элементы ссылаются на значения, в которых они содержатся). Элементы не удаляются из GIN индекса в то время, как значения могут удаляться и изменяться.

В случае, когда список идентификаторов значений индексируемой таблицы мал, он помещается на ту же страницу, что и элемент. В случае, когда такой список велик, используется B-дерево.

GIN индексы чаще всего используются при полнотекстовом поиске или при работе с JSON объектами.

BRIN индекс (от англ. Block Range Index, индекс зон блоков) разбивает таблицу на зоны размером в несколько страниц. Индекс хранит для каждой зоны сводные данные, как правило, минимальное и максимальное значения. При выполнении запроса, если в условии на столбец искомые значения не попадают в диапазон, то вся зона пропускается, если значения попадают, то сканируется вся зона. Структурно BRIN индекс состоит из метастраницы, страниц со сводной информацией о зонах и

обратной карты зон (массив указателей на индексные строки) [3]. Для сканирования индекса используется метод сканирования по битовой карте. Поскольку используются неточные битовые карты, отсюда и неточность самого индекса.

3. Сравнительный анализ индексов

В таблице 1 представлены типы индексов, присутствующих в каждой из рассматриваемых СУБД.

Таблица 1

Таблица оценок индексов в сравниваемых СУБД

Критерий	SQL Server	PostgreSQL
Наличие классического B-tree индекса	+	+
Наличие индекса для работы с XML индексами	Имеет отдельный индекс для XML	Может быть проиндексировано классическим либо GIN индексом
Наличие индекса для работы с JSON объектами	+	+
Наличие индекса для оптимизации запросов с географическими или геометрическими объектами	+	+
Наличие индекса для выполнения полнотекстовых запросов	+	+

Из таблицы 1 можно сделать вывод, что индексы в СУБД SQL Server способны решать те же задачи, что и индексы PostgreSQL. Однако PostgreSQL обладает рядом преимуществ, по сравнению SQL Server.

Классические B-tree индексы присутствуют в обеих СУБД и работают по одному принципу, в случае использования только классических индексов выбор СУБД не имеет значения. Однако, PostgreSQL даёт большой выбор специальных индексов, таких как GiST и SP-GiST индексы для ускорения работы с географическими и геометрическими данными. GiST и GIN индексы в PostgreSQL оптимизируют полнотекстовые запросы (GIN работает в разы быстрее, но дольше строится и занимает больше дискового пространства), BRIN работает с большими таблицами, где использует схему зонирования, GIN индекс удобен для работы с JSON и XML объектами. У SQL Server методов индексной оптимизации гораздо меньше: СУБД имеет отдельный индекс для работы с XML, но для JSON использует классический вариант B-tree индекса, работа с полнотекстовыми запросами в SQL Server реализована, однако не предполагает такого выбора индексов, какой даёт PostgreSQL.

Заключение

В работе был проведён краткий обзор типов индексов в СУБД SQL Server и PostgreSQL.

В ходе анализа было выявлено, что обе СУБД обладают типами индексов, покрывающими основные современные практические задачи при работе с классическими и специальными типами данных. Провести миграцию проекта, содержащего базу данных с индексами на различные, как классические, так и специальные типы данных, представляется возможным. Для старта нового проекта лучшим выбором будет PostgreSQL, так как в данной СУБД больший выбор индексов для оптимизации запросов со специальными типами данных.

Список литературы

1. Нестеров С. А. Корпоративные системы баз данных. – 1-е изд. – СПб: Изд-во Политехн. ун-та, 2014. – 120 с.

2. Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование. – 1-е изд. – Курган: Изд-во Курганского гос. ун-та, 2018. – 178 с.

3. Документация PostgreSQL и Postgres Pro // PostgresPro URL: https://postgrespro.ru/docs (дата обращения: 11.10.2023).

4. Документация по Microsoft SQL // Microsoft learn. – URL: https://learn.microsoft.com/ru-ru/sql/?view=sql-server-ver16 (дата обращения: 10.10.2023).