УДК 004.622

ОПТИМИЗАЦИЯ ВЫБОРКИ ПРОСТРАНСТВЕННЫХ ОБЪЕКТОВ СО СЛОЖНОЙ ТОПОЛОГИЕЙ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

Гимадеев С.В., Васяева Т.А.
Донецкий национальный технический университет

Рассматриваются вопросы оптимизации выборки пространственных данных в системах, работающих с СУБД HBase. Для индексации пространственных данных используются коды Мортона, с помощью которых можно выстраивать иерархическую структуру – дерево квадрантов. Для оптимизации количества индексов используется подход с применением ограничительной рамки (bounding box).

Введение

В последнее время появляется большое количество информационных систем, генерирующих пространственные данные. Это такие системы как сенсорные сети, системы дистанционного зондирования Земли, спутниковый мониторинг транспорта и др. Значительный рост в последнее десятилетие наблюдается в области сенсорных сетей. Этот рост связан с технологиями беспроводных сенсорных сетей (wireless sensor networks). Исследовательская компания Gartner в 2009 году спрогнозировала, что к концу 2012 года сенсорные данные будут создавать 20 % всего не-видео трафика в Internet [1]. Все это говорит о том, что количество и объемы пространственных баз данных будут продолжать быстро расти. Интеллектуальный анализ этих данных может дать много полезной информации о реальном физическом мире и помочь в решении своих задач экспертам таких предметных областей как экология, землеустройство, муниципальное управление, транспорт, экономика и многих других.

Многие современные СУБД уже имеют встроенную поддержку пространственных индексов, что лишний раз говорит о том, что операции с пространственными данными становятся все востребованние. Но многие из подобных систем имеют проблемы с масштабированием, и предпочтение отдается легкомаштабируемым распределенным системам. Для работы с большими объемами данных в последнее время все активнее используется распределенное хранение и обработка. Причем наблюдается переход от использования высокопроизводительных мэйнфреймов к кластерным системам, состоящим из большого числа недорогих серверов с архитектурой x86-64.

После опубликования фирмой Google статей раскрывающей принципы хранения и обработки данных, реализованные ей для своих сервисов [3, 4], появились открытые проекты, реализующие эти принципы. Наиболее успешным из них на сегодняшний день является проект Hadoop.

Hadoop является свободным Java фреймворком, поддерживающим выполнение распределённых приложений, работающих на больших кластерах, объединяющих обычные сервера под управлением открытой операционной системы Linux. Для распределенного хранения данных используется файловая система Hadoop Distributed

File System (HDFS), которая является открытым аналогом Google File System (GFS). Поверх этой файловой системы работает СУБД HBase, которая является аналогом СУБД BigTable от Google [3]. В Hadoop реализована вычислительная парадигма, известная как MapReduce [4].

Согласно этой парадигме приложение разделяется на большое количество небольших заданий, каждое из которых может быть выполнено на любом из узлов кластера. Все эти технологии позволяют достичь очень высокой агрегированной пропускной способности кластера. Эта система позволяет приложениям легко масштабироваться до уровня тысяч узлов и петабайт данных. Поэтому разработку пространственных индексов мы будем выполнять для СУБД HBase [9].

1 Индексирование пространственных объектов

Для построения пространственного индекса мы использовали структуру данных *дерево квадрантов* (quadtree) [5, 6]. Эта структура данных была разработана специально для разбивки двумерного пространства с помощью рекурсивного разделения его на четыре квадранта.

Корень дерева делит область на 4 квадранта, которые затем нумеруются кодами Мортона [7]. Для построения пространственного индекса сначала проводится нормирование координат:

$$x = 180 + longitude; \quad (1)$$

$$y = 90 - latitude, \quad (2)$$

где *longitude* и *latitude* – соответственно географическая долгота и широта в десятичном представлении градусов (без минут и секунд).

Таким образом, мы получим два вещественных числа принимающих значения x [0, 360) и y [0, 180). Затем необходимо преобразовать полученные числа в целые 32-х разрядные. Для этого умножим x и y на 10^7 и отбросим дробную часть. Таким образом, мы максимально используем разрядную сетку 32-х битных переменных с сохранением простоты вычислений. Затем выстраивается код Мортона:

$$m = x_0 + 2y_0 + 2^2 x_1 + 2^3 y_1 + \ldots + 2^{62} x_{31} + 2^{63} y_{31} \quad (3)$$

Таким образом, мы выстроили индекс, способный однозначно определить пространственный объект в одном из уровней дерева квадрантов.

2 Оптимизация выборки объектов со сложной топологией

Описанный выше алгоритм дает возможность сделать точную выборку для такого объекта как точка, если взять сложный полигональный объект, то невозможно построить однозначный индекс.

Для решения данной проблемы, нам необходимо использовать подход, который дал бы возможность проиндексировать сложный объект одним индексом. Такой подход называется ограничивающая рамка (boundin box). Но не будем использовать его в классическом смысле, нам нужно, чтобы ограничивающая рамка строилась не на сам пространственный объект, а для определенного уровня дерева квадрантов, в который попадает объект.

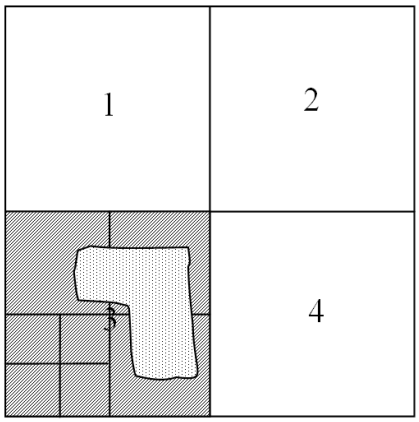


Рисунок 1. Разбиение на сектора

Таким образом, за одну выборку мы сможем получить все пространственные объекты, которые находятся в данном узле дерева. Далее следует преобразовать выбранные данные в геометрические объекты и выбрать необходимый объект по предикату отношения равенства (equals) [9]. Для поиска пространственных объектов можно использовать и другие предикаты, например, предикат пересечения (intersects) [9] для выборки всех объектов которые пересекаются с искомым объектам.

3 Оценка производительности

Пространственные запросы обычно подразумевают выполнение следующих действий:

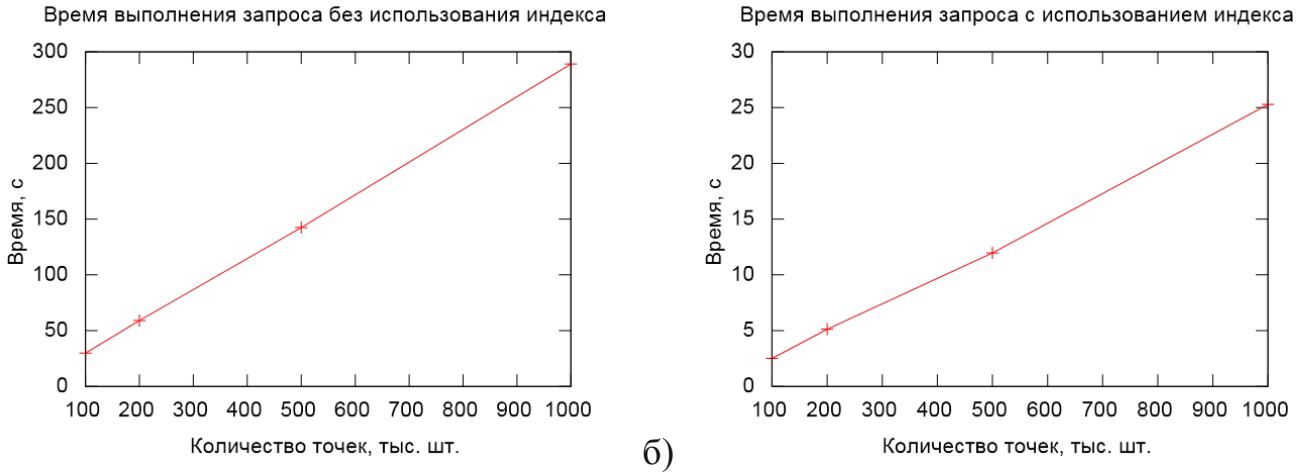
1. Поиск объектов, вложенных в заданный многоугольник. Найденные объекты должны целиком помещаться внутри заданной области.
2. Поиск объектов, пересекающихся с заданным многоугольником. Найденные объекты должны полностью или хотя бы частично помещаться внутри заданной области.

С учетом этого были проведены экспериментальные исследования с использованием системы Hadoop (CDH3 Beta 4), работающей в автономном режиме на сервере (4 ядра, 2GB RAM) под управлением Ubuntu Linux с Sun JRE 1.6.0_24.

Случайным образом генерировался миллион пространственных объектов и выполнялось построение индексов. Затем случайно генерировалась искомая область и выбирались объекты полностью или частично попадающие в искомую область. Тест проводился много раз и результирующие данные являются усредненными для всех этих тестов. Результаты теста представлены на рис. 2.

Выводы

Информационные системы, работающие с пространственными данными, становятся все более актуальными. Для распределенного хранения и обработки пространственных данных мы использовали технологии, применяемые при построении современных высоконагруженных веб-систем. Были построены индексы для ускорения выполнения операций выборки объектов принадлежащих определенной территории и определения отношений соседства между пространственными объектами. Эксперименты показали значительное уменьшение времени выполнения пространственных запросов. Данный



а) б)

Рисунок 2. Влияние использования индексов
на время выполнения пространственных запросов

подход может использоваться также и при разработке алгоритмов интеллектуального анализа пространственных данных [9].

Перечень использованных источников

[1] Gartner. Gartner Predicts Video Telepresence Will Replace 2.1 Million Airline Seats Per Year by 2012. http://www.gartner.com/it/page.jsp?id=876512.

[2] Ester M., Kriegel H.-P., Sander J. Knowledge Discovery in Spatial Databases, invited paper at 23rd German Conf. on Artificial Intelligence (KI '99), Bonn, Germany, 1999.

[3] Chang F., Dean J., Ghemawat S., et al. Bigtable: a distributed storage system for structured data. In Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation - Volume 7 (OSDI '06), Vol. 7. USENIX Association, Berkeley, CA, USA, 15-15.

[4] Dean J., Ghemawat, S. Mapreduce: simplified data processing on large clusters. Commun. ACM, 51:107–113, January 2008.

[5] Samet H. The Design and Analysis of Spatial Data Structures. Addison-Wesley, Reading, MA, 1990.

[6] Finkel R., Bentley J.L. Quad Trees: A Data Structure for Retrieval on Composite Keys. Acta Informatica 4 (1): 1–9, 1974.

[7] Morton G.M. A computer oriented geodetic data base and a new technique in file sequencing. Technical report, IBM Ltd., Ottawa, Ontario, Mar. 1966.

[8] Herring J.R. OpenGIS Implementation Standard for Geographic information – Simple feature access – Part 1: Common architecture. Technical report, OGC, 2010. (Имеется русский перевод издания 2006 года: http://gis-lab.info/docs/ogc-sfa1-v1.pdf).

[9] А.О. Телятников, С.В. Гимадеев «Индексация пространственных данных для ускорения их интеллектуального анализа» http://masters.donntu.edu.ua/2012/fknt/gimadeev/library/article1.pdf