Какие методы и технологии используются для обработки Больших Данных

Текст Михаил Цымблер

Для аналитической обработки Больших Данных используется широкий спектр методов и алгоритмов. Это методы классов Data Mining (поиск ассоциативных правил, классификация, кластеризация и др.) и Machine Learning, искусственные нейронные сети и распознавание образов, имитационное моделирование, статистический анализ и др.

Наиболее часто в качестве аппаратной платформы для систем обработки Больших Данных используются многопроцессорные вычислительные системы без совместного использования ресурсов (Shared Nothing Architecture), которые могут обеспечить массивно-параллельную обработку данных, масштабируемую без деградации на сотни и тысячи узлов.

Существует также ряд аппаратно-программных решений, ориентированных на обработку Больших Данных. Данные решения представляют собой готовые к установке в ЦОД телекоммуникационные шкафы, содержащие кластер серверов и управляющее программное обеспечение для массово-параллельной обработки. Наиболее известными аппаратно-программными комплексами для обработки Больших Данных являются Aster MapReduce Appliance корпорации Teradata, Big Data Appliance корпорации Oracle, Greenplum Appliance корпорации EMC. Существуют также комплексы для аналитической обработки в оперативной памяти: система Hana компании SAP и система Exalytics корпорации Oracle, построенная на основе реляционной СУБД TimesTen и многомерной СУБД Essbase.

В список наиболее популярных технологий обработки Больших Данных в настоящее время, по-видимому, входят NoSQL, MapReduce и Hadoop, а также параллельные СУБД.

Термин NoSQL (Not Only SQL — не только SQL) обозначает набор подходов к реализации хранилищ данных, которые основаны на модели данных, отличной от реляционной. Интерес к технологиям NoSQL молниеносно возник после того, как корпорация Google в начале 2000-х опубликовала документацию о распределенной файловой системе BigTable, которая способна обработать 20 Петабайт информации в день и является базисом поисковой системы Google и популярных сервисов Gmail, Google Maps, Google Earth и др. Особенностью NoSQL-технологий является идея неограниченного масштабирования и отказ от согласованности данных в угоду производительности их обработки. Системы класса NoSQL проектируются с расчетом на неограниченное горизонтальное масштабирование: добавление или удаление узлов в кластере не должно сказаться на работоспособности системы.

В отличие от реляционных СУБД, NoSQL системы не поддерживают транзакции с ACID-свойствами (Atomicity — атомарность, Consistency — согласованность, Isolation — изолированность, Durability — долговременность).

В силу отсутствия поддержки транзакций NoSQL–решения не могут использоваться в приложениях, где указанные свойства являются необходимостью, например, в системах обслуживания бирж и банков.

Однако в системах обслуживания запросов многомиллионной web-аудитории пользователей свойства ACID, несмотря на всю их привлекательность, обеспечить практически невозможно. Поэтому в NoSQL-системах, обрабатывающих Большие Данные, согласованностью данных жертвуют ради достижения двух других свойств: до-

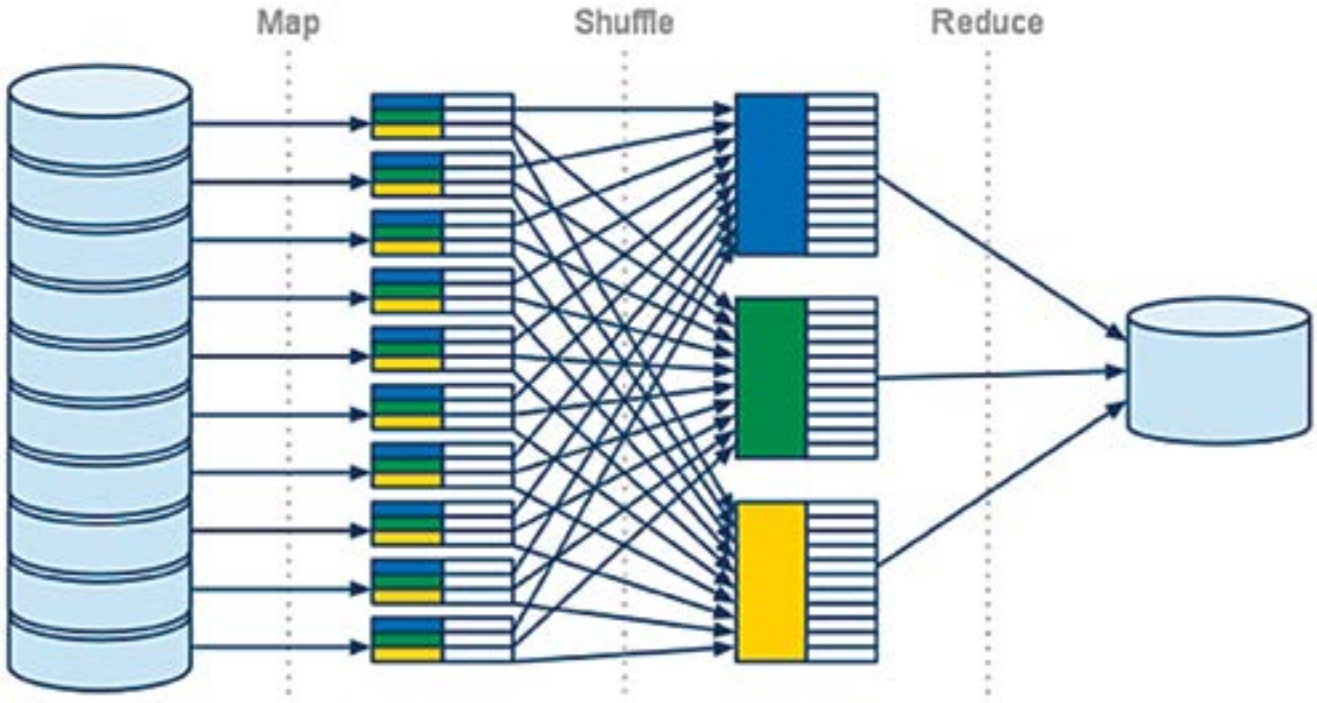

Рис. 1. Парадигма MapReduce

ступность данных и устойчивость к разбиению данных по узлам вычислительной системы. Подобная особенность распределенных вычислений — возможность обеспечить не более двух свойств из трех, указанных выше, — известна как теорема CAP (аббревиатура от англоязычных названий свойств: Consistency, Availability, Partition tolerance), хотя, строго говоря, таковой не является в силу отсутствия четкой формальной постановки задачи.

На сегодня существует большое количество разнообразных NoSQL-систем. В соответствии с классификацией портала nosql-database.org различают хранилища «ключ-значение», документ-ориентированные системы, колоночные хранилища и системы обработки графов.

Хранилища «ключ-значение» строятся на основе ассоциативных массивов, позволяющих работать с данными по ключу; какая-либо информация о структуре значений не сохраняется. Такие хранилища используются в основном для хранения изображений, создания специализированных файловых систем, в качестве кэшей для объектов. Примеры систем «ключ-значение»: Redis, Scalaris, Riak, DynamoDB и др.

Документ-ориентированные системы предназначены для хранения иерархических структур данных (документов). Документы могут быть сгруппированы в коллекции, причем коллекции могут содержать другие коллекции. Системы данной группы применяются в системах управления содержимым, издательском деле, документальном поиске и др. Примеры систем: MongoDB, Berkeley DB XML, SimpleDB, CouchDB и др.

Колоночные хранилища основаны на идее хранения данных на диске не по строкам, как это делают реляционные СУБД, а по колонкам. С точки зрения SQL-клиента, данные представлены в виде таблиц, однако физически эти таблицы являются набором колонок, каждая из которых представляет собой таблицу из одного поля. При этом физически, на диске, значения одного поля хранятся последовательно. При выполнении выборки данных такая организация хранения обеспечивает

В настоящее время практически любая международная научная конференция, тематика которой предполагает обработку данных, в своем информационном сообщении обязательно указывает «Big Data» в списке тем принимаемых статей. Крупные IT-корпорации создают научные центры для исследований в области Больших Данных

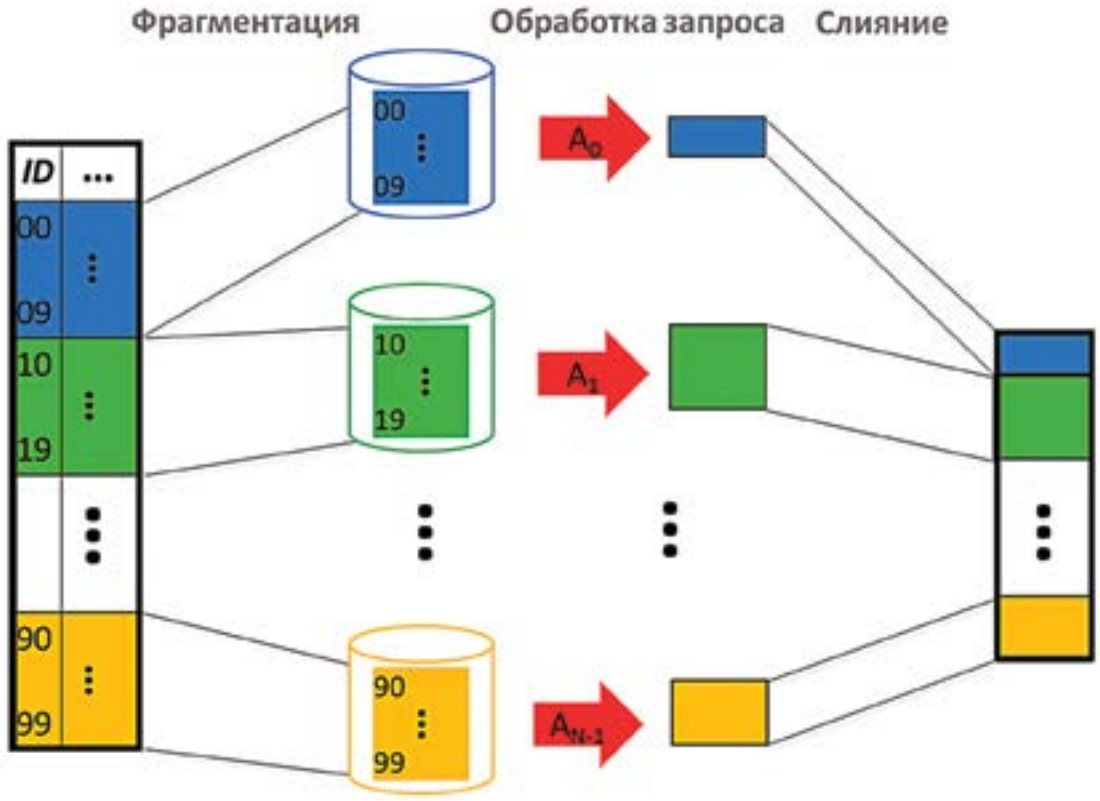

Рис. 2. Фрагментный параллелизм

сокращение времени обработки за счет выполнения чтения только тех полей, которые требуются в запросе. Примеры колоночных хранилищ: BigTable, Hbase, Cassandra и др.
Системы обработки графов применяются для работы с данными, которые естественным образом представляются в виде графов (социальные и биологические сети, WWW, потоки работ и др.). Модель данных включает в себя понятия вершин, ребер и свойств. Работа с данными осуществляется путем обхода графа по ребрам с заданными свойствами. Примеры систем обработки графов: Neo4j, Pregel, Giraph, Pegasus, Titan и др.
Парадигма распределенных вычислений MapReduce разработана корпорацией Google в 2004 г.
В рамках данной парадигмы (см. рис. 1) один из узлов кластерной системы трактуется как мастер, остальные — рабочие.
Обработка данных выполняется в виде последовательности из двух шагов: Map и Reduce. На шаге Map узел-мастер получает входные данные и распределяет их по узлам-рабочим. На шаге Reduce узел-мастер выполняет свертку данных, предварительно обработанных рабочими, и отправку конечного результата пользователю. Широко известен эксперимент корпорации Google, в ходе которого производилась сортировка 1 Петабайта данных при помощи фреймворка MapReduce. Данные были представлены в виде 10 трлн записей размером 100 байт каждая. Кластер из 4000 компьютеров выполнил сортировку беспрецедентного для такого типа задач объема данных за 6 часов 2 минуты.
Hadoop — проект фонда Apache Software Foundation, который представляет собой фреймворк для разработки и выполнения распределенных программ, работающих на кластерах из сотен и тысяч узлов. Используется для реализации высоконагруженных веб-сайтов (например, Yahoo! и Facebook), разработан на Java в рамках вычислительной парадигмы MapReduce. Hadoop состоит из следующих частей: набор инфраструктурных программных библиотек и утилит Hadoop Common, распределенная файловая система HDFS, система для планирования заданий и управления кластером YARN, а также платформа программирования и выполнения распределенных MapReduce-вычислений Hadoop MapReduce.
С Hadoop ассоциирован ряд проектов и технологий, многие из которых изначально развивались в рамках этого проекта, но впоследствии стали самостоятельными.
Распределенное хранилище данных Hive обеспечивает управление данными в HDFS и предоставляет язык запросов HiveQL, основанный

на SQL. Запросы HiveQL транслируются в MapReduce-задачи. Pig — среда исполнения и высокоуровневый язык для описания вычислений в Hadoop. Программы Pig также транслируются в MapReduce-задачи. Параллельные СУБД, в противовес NoSQL-решениям, не отказываются от реляционной модели данных. Базовой концепцией параллельных СУБД является фрагментный параллелизм (см. рис. 8), который подразумевает горизонтальную фрагментацию каждой таблицы базы данных по дискам кластерной системы. Способ фрагментации определяется функцией фрагментации, которая для каждой записи таблицы вычисляет номер вычислительного узла кластера, где должна храниться данная запись. На каждом узле кластерной системы запускается параллельный агент (ядро СУБД), обрабатывающий запросы пользователей. Один и тот же запрос параллельно выполняется каждым агентом над «своими» фрагментами таблиц базы данных, и затем полученные частичные результаты сливаются в результирующую таблицу. Несмотря на независимую обработку агентами «своих» фрагментов базы данных, для получения корректного результата необходимо выполнять пересылки записей во время выполнения запроса (при выполнении операции соединения двух таблиц по общей колонке). Для организации таких пересылок выполняется распараллеливание последовательного плана запроса: в дополнение к обычным реляционным операциям (соединение, выборка, проекция и др.) в нужные места плана запроса вставляется специальная операция exchange, которая обеспечивает пересылку «чужих» записей соответствующим параллельным агентам и прием «своих» записей от них, не нарушая хода выполнения запроса. В силу того, что различные фрагменты таблицы могут иметь существенно различные размеры (такая ситуация носит название «Перекос данных»), в параллельных СУБД для балансировки загрузки параллельных агентов используется техника частичной репликации базы данных. К классу параллельных СУБД можно отнести системы Greenplum, Netezza, EXASOL и др.

Следует еще сказать о взаимопроникновении технологий параллельных СУБД и MapReduce. Например, система HadoopDB представляет собой архитектурный гибрид парадигмы MapReduce и технологий реляционных СУБД. В качестве фреймворка, реализующего MapReduce-вычисления, в СУБД HadoopDB используется система Hadoop. Hadoop обеспечивает коммуникационную инфраструктуру, объединяющую узлы кластера, на которых выполняются экземпляры СУБД PostgreSQL. Запросы пользователя на языке SQL транслируются в задания для среды MapReduce, которые далее передаются в экземпляры СУБД. В системах Greenplum и nCluster модель MapReduce реализуется внутри СУБД, и возможностями этих реализаций могут пользоваться разработчики аналитических приложений: в Greenplum Database — наряду с SQL, а в nCluster — из SQL.

Упомянем также библиотеки и фреймворки, используемые для интеллектуального анализа Больших Данных. Apache Mahout представляет собой свободную библиотеку алгоритмов машинного обучения с открытым кодом. Реализация библиотеки выполнена на языке Java, масштабируемость достигается за счет использования фреймворка Hadoop. Инструментарий, предоставляемый библиотекой, в настоящий момент позволяет реализовать рекомендательные системы, кластеризацию и классификацию Больших Данных.

R — язык программирования для статистической обработки данных и работы с графикой, а также свободный фреймворк с открытым исходным кодом, включающий в себя более 4000 пакетов статистических и аналитических функций для различных областей применения. Вышеупомянутый аппаратно-программный комплекс Big Data Appliance корпорации Oracle содержит интегрированные программные средства на основе языка R и фреймворка Hadoop; поддержка языка R встроена также в СУБД Oracle и Netezza. Свободные фреймворки RHIPE и RHadoop позволяют интегрировать возможности языка R и фреймворка Hadoop для аналитической обработки Больших Данных.

Большие Данные: что будет дальше?

За время, которое вы потратили на чтение этой статьи, мировые Большие Данные приросли как минимум десятком-другим терабайт. Можно констатировать, что Больших Данных меньше не станет. Поэтому единственный выход в данной ситуации для научного и бизнес-сообщества — активный поиск новых методов, технологий и аппаратных решений для решения проблем, связанных с обработкой Больших Данных.

В соответствии с этим в ряде западных вузов (например, в Вашингтонском университете, Университете Калифорнии в Беркли и в Нью-Йоркском университете) открываются учебные программы в области data science (науки о данных). В настоящее время практически любая международная научная конференция, тематика которой предполагает обработку данных, в своем информационном сообщении обязательно указывает «Big Data» в списке тем принимаемых статей. Крупные IT-корпорации создают научные центры для исследований в области Больших Данных, например, Big Data Science and Technology Center — совместный центр корпорации Intel и Массачусетского технологического института.