

ЗАДАЧА ОПТИМИЗАЦИИ РАБОТЫ КЛАСТЕРА С ПОМОЩЬЮ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ



К.Ю. Слисенко¹

Старший разработчик в компании JazzTeam, магистр технических наук, Республика Беларусь



С.С. Божко²

Магистрант кафедры Информатики, разработчик в Adform, Республика Беларусь



С.И. Сиротко³

Доцент, кандидат физико-математических наук, кафедра Информатики

¹Белорусский государственный университет информатики и радиоэлектроники, kislisenko@gmail.com

²Белорусский государственный университет информатики и радиоэлектроники, svt.bozhko@gmail.com

³Белорусский государственный университет информатики и радиоэлектроники, sergeyis@tut.by

Article describes the problem of Big Data cluster performance optimization. This work was done at BSUIR during Big Data Predictive Analytics course project, addressing problem formulated by Dr. Boris Zibitsker from BEZNext (www.beznext.com). As an example we used BEZNext virtual lab with small clusters of various systems, including: Hadoop, Cassandra, etc. Article proposes how to use machine learning algorithms for building automatic control and management system for Big Data clusters.

Современные технологии позволяют проводить аналитические исследования больших объёмов данных. Задачи анализа задач можно условно разбить на следующие категории [1]:

Descriptive analytics позволяет понять что случилось и когда это произошло;

Diagnostic analytics позволяет выявить сбои, проблемы и их причины;

Predictive analytics позволяет предсказать работу процесса в случае изменения каких-либо факторов;

Prescriptive analytics даёт автоматические советы как улучшить процесс и не допустить проблем и сбоев;

Control analytics автоматически управляет процессом на основе выработанных на этапе анализа рекомендаций.

Эти категории можно сопоставить с любой бизнес областью и предложить идеи инновационных продуктов.

Вычислительный кластер – это множество машин, объединённых в общую сеть и предоставляющих конечный объём разделяемых ресурсов. В зависимости от типов приложений может существенно различаться характер нагрузки: некоторые приложения активно используют CPU, другие – активно выполняют чтение и запись на диск и так далее. Объектом исследования является вычислительный кла-

стер, содержащий сконфигурированные программные системы обработки и хранения данных. Кластер состоит из realtime и batch-составляющих. Realtime-часть представлена двумя машинами с установленными Kafka, Storm, Zookeeper[2]. Batch-часть представлена тремя машинами с установленным менеджером ресурсов YARN, и фреймворками Hadoop и Tez[2]. Одна из них является мастером и содержит HDFS name node, Spark master. Две другие содержат HDFS data node, Spark worker и базу данных Cassandra[2].

Как batch, так и realtime составляющие активно используются при обработке и анализе данных, поэтому для исследования очень важно наличие этих обоих составляющих.

YARN – это менеджер ресурсов, поверх которого работают системы параллельной обработки и хранения данных.

HDFS – распределённая файловая система с возможностью быстрой последовательной записи и чтения, а так же с механизмами репликации и защиты от сбоев.

Cassandra – распределённая NoSQL база данных, оптимизированная под запись.

Hadoop, Spark, Tez – системы параллельной обработки данных по принципу Map-Reduce[3] и DAG[2]. Storm – система потоковой обработки данных. Kafka – система приёма и хранения сообщений.

Для исследования характеристик кластера производится запуск приложений, тестирующих производительность. Их описание представлено в таблице 1.

Таблица 1 – Приложения для тестирования производительности кластера

Система	Приложение	Описание
Hadoop	Terasort	Выполняет сортировку одного терабайта данных. Могут быть взяты как готовые данные, так и специально сгенерированные.
Hadoop	Word count	Подсчёт количества слов в текстовых файлах.
Tez	Ordered word count	Подсчёт количества слов в текстовых файлах с сортировкой.
Spark	Spark PI	Вычисление числа Пи методом аппроксимации.
Storm	Word count	Подсчёт количества слов в потоке данных.
Cassandra	Stress tool	Вставка и чтение большого объёма записей в базы данных.

Для осуществления замеров используемых ресурсов на каждой машине работает программный агент, который с заданной периодичностью сохраняет состояние данной машины в лог-файлы. Примеры собранной информации представлены в таблицах 2 и 3.

Таблица 2 – Пример собранной агентом информации о процессах и пользователях

Timestamp	User	Process id	Command
1417609431	user1	931	java -classpath ... -jar storm.jar backtype.storm.ui.core
1417609431	user2	1340	java -classpath ... org.apache.spark.deploy.worker

Таблица 3 – Пример собранной агентом информацией о процессах и ресурсах

Timestamp	Process id	Name	Average CPU	Read bytes	Write bytes
1417609431	931	java	10.5	450123	1240
1417609431	1340	java	25.3	344	679034

Данные об использовании ресурсов переносятся ETL-процессами в NoSQL базу данных Cassandra. Схема ETL-процесса представлена на рисунке 1.

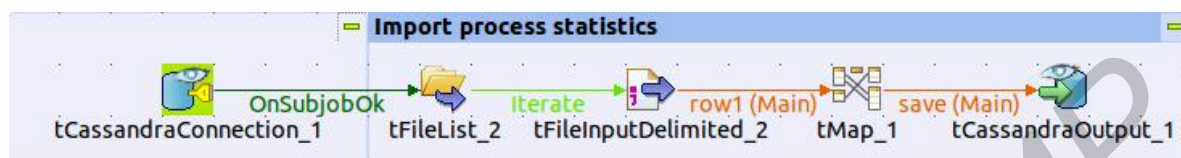


Рис. 1. Схема ETL-процесса переноса данных в Cassandra

Алгоритмы подсчёта статистики и машинного обучения выполняются системой Spark. Результаты работы сохраняются в Cassandra. Для визуализации результатов используется система Spago BI, отображающая данные в виде графиков и диаграмм. Архитектура разрабатываемой аналитической системы представлена на рисунке 2.

Применив описанный выше спектр аналитических задач к специфике вычислительного кластера, можно получить описание возможной аналитической системы. Ниже представлены некоторые идеи и возможные варианты их реализации.

В рамках descriptive analytics будет реализовано построение графиков использования ресурсов, выявление сезонности и пиков. Необходимо выделить отдельные приложения — совокупности процессов, логически объединённых с точки зрения пользователя. Статистика должна быть посчитана целиком для каждого приложения.

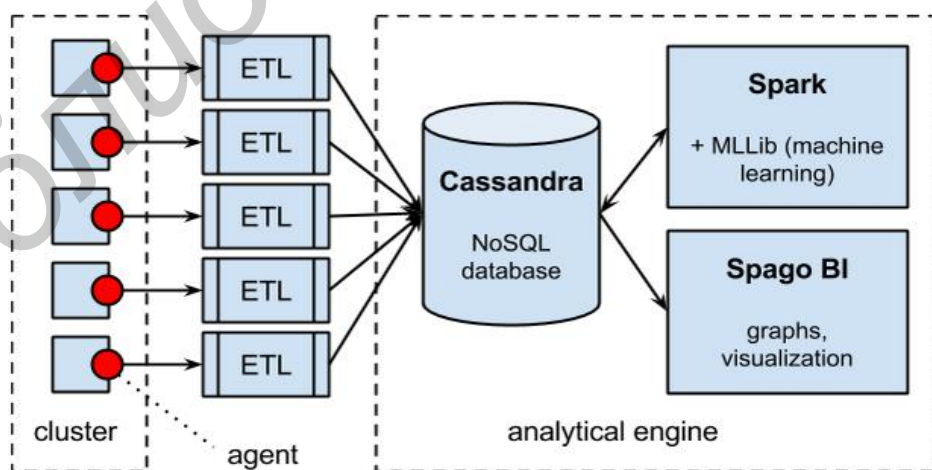


Рис. 2. Архитектура аналитической системы

Diagnostic analytics будет выявлять отклонения от привычной работы приложений. Должна быть возможность определить чрезмерное потребление ресурсов, а так же приложения и пользователей, которые могут быть этому причинами.

В рамках predictive analytics необходимо построить модель для прогноза работы кластера в случае изменения параметров: увеличения количества данных, пользователей, приложений, изменения конфигурации кластера. Модель также позволит предположить, какое минимальное количество ресурсов следует добавить, чтобы соответствовать определённым показателям производительности. С помощью модели можно будет спрогнозировать, не скажется ли существенно изменения в конфигурации одного приложения на работу других.

В рамках prescriptive analytics необходимо кластеризовать приложения для выделения схожих по характеру потребления ресурсов. На основе этой информации будут формироваться автоматические советы по выделению ресурсов кластера в зависимости от сложившейся ситуации. Может быть выполнен перебор различных параметров в predictive-модели для поиска оптимальных приоритетов выполнения.

Control analytics позволит осуществить автоматическое выделение ресурсов и приоритезацию задач. В результате должна получиться система управления кластером. Предполагаемая схема такой системы представлена на рисунке 3.

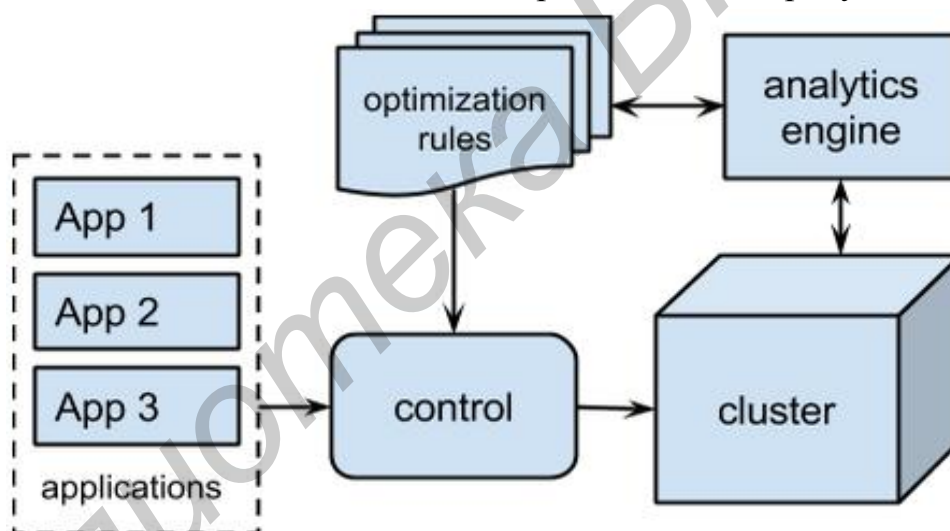


Рис. 3. Схема системы управления кластером с применением control analytics

Важным аспектом работы системы является поддержание performance management процесса[1]. Необходимо выполнять постоянное сравнение прогнозных данных с реальными и уточнять модель для достижения лучших результатов.

Литература

1. Managing Workload Performance on a Private Cloud. [Электронный ресурс]. – Режим доступа: http://www.beznex.com/assets/files/Managing_workloads_on_Private_Cloud_WP.pdf. – Дата доступа: 16.05.2015.
2. Grover M., Malaska T. // Hadoop application architectures // O'Reilly, 2014.
3. Gillick, D. // MapReduce: Distributed Computing for Machine Learning // Berkeley University. 2006.