Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК

Мялик
Илья Сергеевич

Программно-алгоритмическое обеспечение организации процесса обработки большого потока данных в информационно-вычислительной среде

АВТОРЕФЕРАТ
на соискание академической степени
магистра технических наук

по специальности 1-40 80 05 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

Научный руководитель
Куликов С.С.
к.т.н., доцент

Минск 2019

КРАТКОЕ ВВЕДЕНИЕ

За последнее десятилетие мы увидели немало интересных нововведений и усовершенствований в сферах баз данных (БД), распределенных систем, а также в способах создания работающих с ними приложений. Вот некоторые факторы, приведшие к этим усовершенствованиям.

Такие интернет-компании, как Google, Yahoo!, Amazon, Facebook, LinkedIn, Microsoft и Twitter, обрабатывают колоссальные объемы данных и трафика, что вынуждает их создавать новые инструменты, подходящие для эффективной работы в подобных масштабах.

- Свободное программное обеспечение стало чрезвычайно популярным и во многих случаях является более предпочтительным по сравнению с коммерческим ПО и ПО для внутреннего использования.
- Тактовые частоты процессоров не слишком возросли, но многоядерные процессоры стали стандартом, плюс увеличились скорости передачи данных по сети. Это означает дальнейший рост параллелизма.
- Даже небольшая команда разработчиков может создавать системы, распределенные по множеству машин и географических регионов, благодаря такой IaaS, как Amazon Web Services.
- Многие сервисы стали высокодоступными; длительные простои из-за перебоев в обслуживании или текущих работ считаются все менее приемлемыми.

Высоконагруженные данными приложения открывают новые горизонты возможностей благодаря использованию этих технологических усовершенствований. Мы говорим, что приложение является высоконагруженным данными, если те представляют основную проблему, с которой оно сталкивается, – качество данных, степень их сложности или скорость изменений, – в отличие от высоконагруженных вычислений, где узким местом являются циклы CPU.

Цель научной работы – выбрать методы и построить систему для обработки и хранения данных. Для этого будут рассмотрены внутренние устройства этих систем, разбор ключевых алгоритмов, выявлены их принципы и неизбежные компромиссы. Будут выведены удобные подходы к информационным системам.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Цель и задачи исследования

Целью диссертационной работы является определение наиболее выгодной архитектура для проектирования высоконагруженной системы, выбора ключевых компонентов и узлов для обработки данных, а также интеграции программно-алгоритмического обеспечения организации процесса обработки большого потока данных в контекст выбранной архитектуры.

Для достижения поставленной цели необходимо решить следующие задачи:

1. определить подходы и практики для проектирования высоконагруженных архитектур;
2. выбрать ключевые компоненты для каждого из улов проектируемой системы;
3. провести анализ программно-алгоритмических обеспечений для обработки большого потока данных;
4. проинтегрировать программно-алгоритмическое обеспечений в выбранную архитектуру для высоконагруженных систем;
5. собрать результирующие данных для определения работоспособности построенной структурной модели.

Объектом исследования являются высоконагруженные системы и большие данные.

Предметом исследования является пути построения высоконагруженных систем и программно-алгоритмическое обеспечение организации процесса обработки большого потока данных в информационно-вычислительной среде.

Основной *гипотезой*, положенной в основу диссертационной работы, является определение наиболее выгодной архитектура для проектирования высоконагруженной системы, выбора ключевых компонентов и узлов для обработки данных, а также интеграции программно-алгоритмического обеспечения организации процесса обработки большого потока данных в контекст выбранной архитектуры. Также главным фактором написание данной научной работы служит поиск пути определения наилучшей архитектуры и сопутствующих компонентов их всего множество уже разработанных ранее.

Личный вклад соискателя

Результаты, приведенные в диссертации, получены соискателем лично и вкупе содержат только научные выдержки, приобретенные за время обучения в магистратуре. Вклад научного руководителя С. С. Куликова, заключается в формулировке целей и задач исследования и в поиске дополнительной литературы в белорусских и зарубежных источниках, а также помогал с переводом некоторых из них.

Опубликованность результатов диссертации

По теме диссертации опубликовано 2 печатные работы, из них 1 статья в рецензируемом издании, 1 из работ в сборниках трудов и материалов конференций.

Структура и объем диссертации

Диссертация состоит из введения, общей характеристики работы, трех глав, заключения, списка использованных источников, списка публикаций автора и приложений. В первой главе представлены методы и практики проектирования высоконагруженных система. Вторая глава посвящена программным средствам для обеспечения обработки данных. В третьей главе построение архитектуры высоконагруженный системы и определение эффективного метода обработки данных.

Общий объем работы составляет 69 страниц, из которых основного текста – 40 страниц, 20 рисунков на 10 страницах, 7 таблиц на 9 страницах, список использованных источников из 46 наименований на 5 страницах и 3 приложения на 5 страницах.

ОСНОВНОЕ СОДЕРЖАНИЕ

Во **введении** определена область и указаны основные направления исследования, показана актуальность темы диссертационной работы, дана краткая характеристика исследуемых вопросов, обозначена ценность работы.

В первой главе было выявлено что при проектировании системы для обработки большого объёма данных стоит учитывать множество особенностей, а именно, обеспечение надёжности системе, что бы она могла обрабатывать данные всегда и без перебоев. Обеспечение масштабируемости, при которой система может уменьшать/увеличивать количество информационно-вычислительных узлов автоматически. Так же не стоит забывать и про удобство сопровождения.

Рисунок 1 – Алгоритм репликации с ведущим узлом

Не малую роль при обработке большого объёма данных несет их хранение. А для безлопастного и надежного хранения данных стоит использовать такие подходы как: репликация и секционирование.

Сама по себе потоковая обработка данных, мало чем отличается от пакетной, разве что данные всегда идут одним непрекращающимся потоком(и). Для того что бы их обработать данные все равно надо разбивать, во много может помочь брокер сообщение, который будет разбивать поток на сообщения, которые, в последствии, будут обработаны. Не стоит забывать и про методы объедения похожих по типу потоков данных в один для более удобной и просто обработки.

Во второй главе были проанализированы возможные пути решения проблем обработки большого объёма данных, были выявлены концептуальные ограничения, которые не позволяют классической реляционной архитектуре справляться со стремительно возрастающим объемом информации.

Далее были рассмотрены три подхода к работе с большими данными: NoSQL, MapReduce и обработка потоков событий в реальном времени. Важно отметить, что ни один из представленных подходов не предлагает решения всех возможных задач, которые возникли в контексте больших данных. Каждый из них эффективно решает свой класс задач.

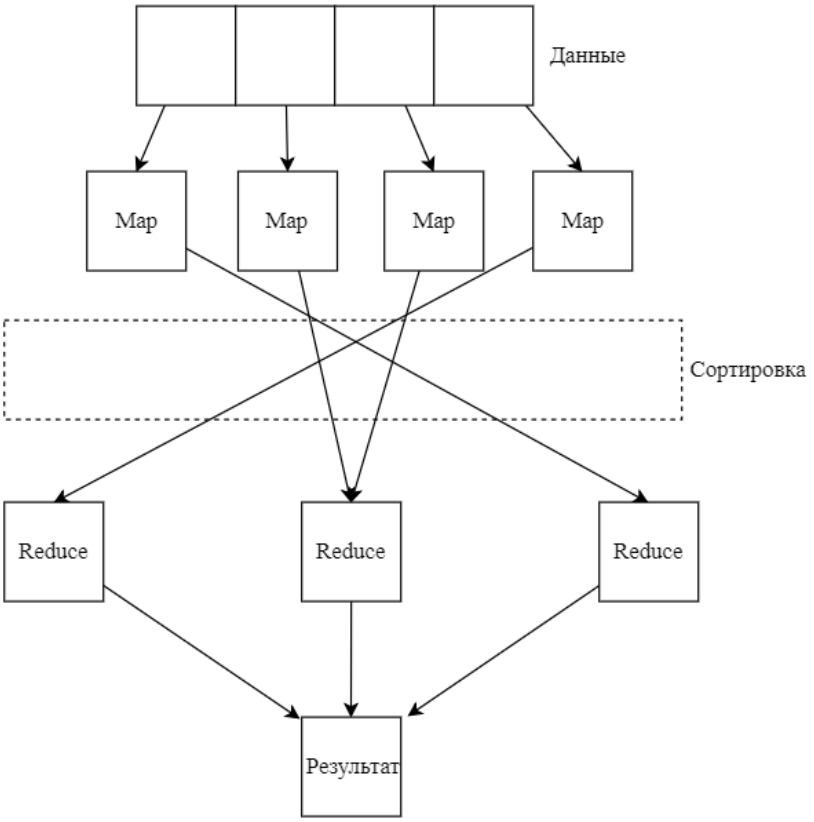

Рисунок 2 – Схема MapReduce вычислений

В третей главе была рассмотрена архитектура MapReduce, а также рассмотрены особенности работы с использованием инструмента Pig. Было определено что написание программ для MapReduce представляет собой весьма непростую задачу, требующую особого подхода. Все сложности, однако, компенсируются мощностью, возможностями масштабирования и высокой скоростью обработки огромных объемов произвольных данных.

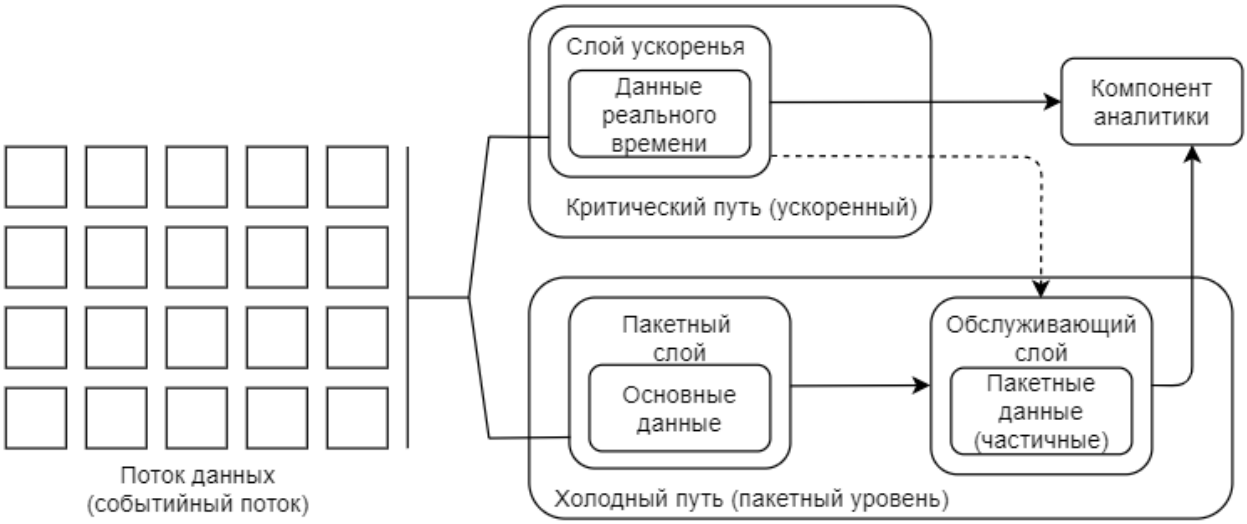

Рисунок 3 – Лямбда-архитектура для обработки большого объёма данных

В экспериментальная части были рассмотрены такие архитектуры для обработки больших данных как лямбда-архитектура и каппа-архитектура. В контексте данной задачи были выявлены все плюсы и минусы. Для обработки большого объёма данных как компонента потоковой/пакетной обработки было выбрано программное средство Pig. Как результат экспериментальной части были получены данные применения архитектуры для построения высоконагруженных систем в комбинации с компонентом Pig.

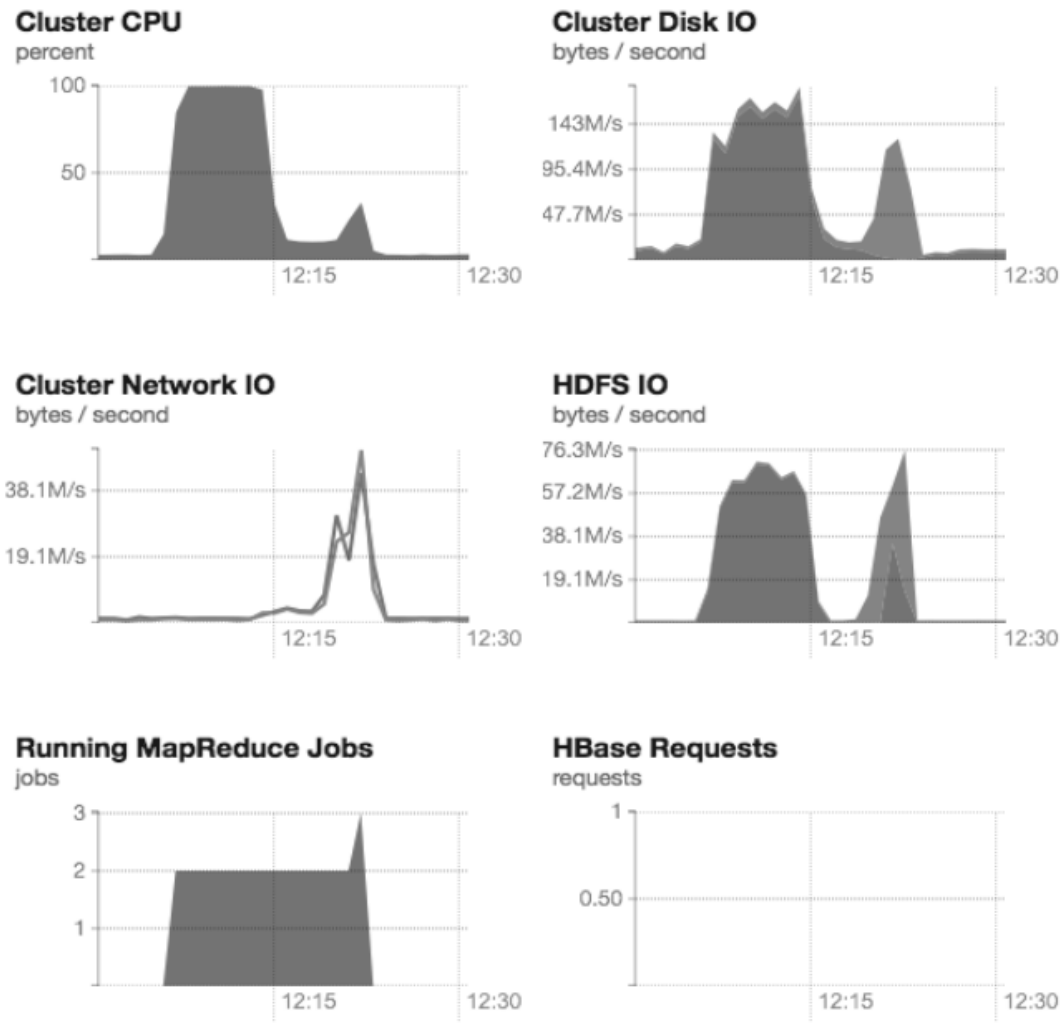

Рисунок 4 – Результаты работы системы

ЗАКЛЮЧЕНИЕ

В научной работе был проведен анализ компонентов высоконагруженных систем, используемых в контексте проектирования приложений для обработки большого объёма данных. Так же были рассмотрены основные критерии обеспечения надежности, масштабируемости и удобства сопровождения систем.

Проведя выборочное исследования, были найдены подходы для внедрения методов хранения данных путем распределения кластеров по контекстным доменам, что включает в себя внедрение репликации и секционирование.

В области потоковой обротки данных в контексте высоконагруженных систем были рассмотрены очереди и брокеры сообщений, а также методы объединения потоков данных вида: поток-поток, таблица-поток, таблица-таблица.

Было проведено сравнение реляционных СУБД и NoSql баз данных как хранилищ для больших данных. Так же были рассмотрены методы и подходы обработки данных с помощью MapReduce и Pig.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

1. Мялик, И. С. Анализ побочных данных состояния системы и данных результирующих точек системы / И. С. Мялик // Компьютерные системы и сети: материалы 54-й научной конференции аспирантов, магистрантов и студентов / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2018. – С. 95 – 96.

2. Мялик И.С. Репликация как один из способов обеспечения надежности системы // Интернаука: научный журнал. № 21(103). – М., Изд. «Интернаука», 2019. – С. 29 – 30