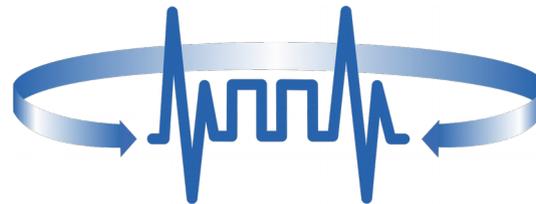


Сервис-ориентированная инфраструктура для обучения технологиям обработки больших данных

Сухорослов О.В.

Институт проблем передачи информации РАН
Школа анализа данных, НИУ Высшая школа экономики



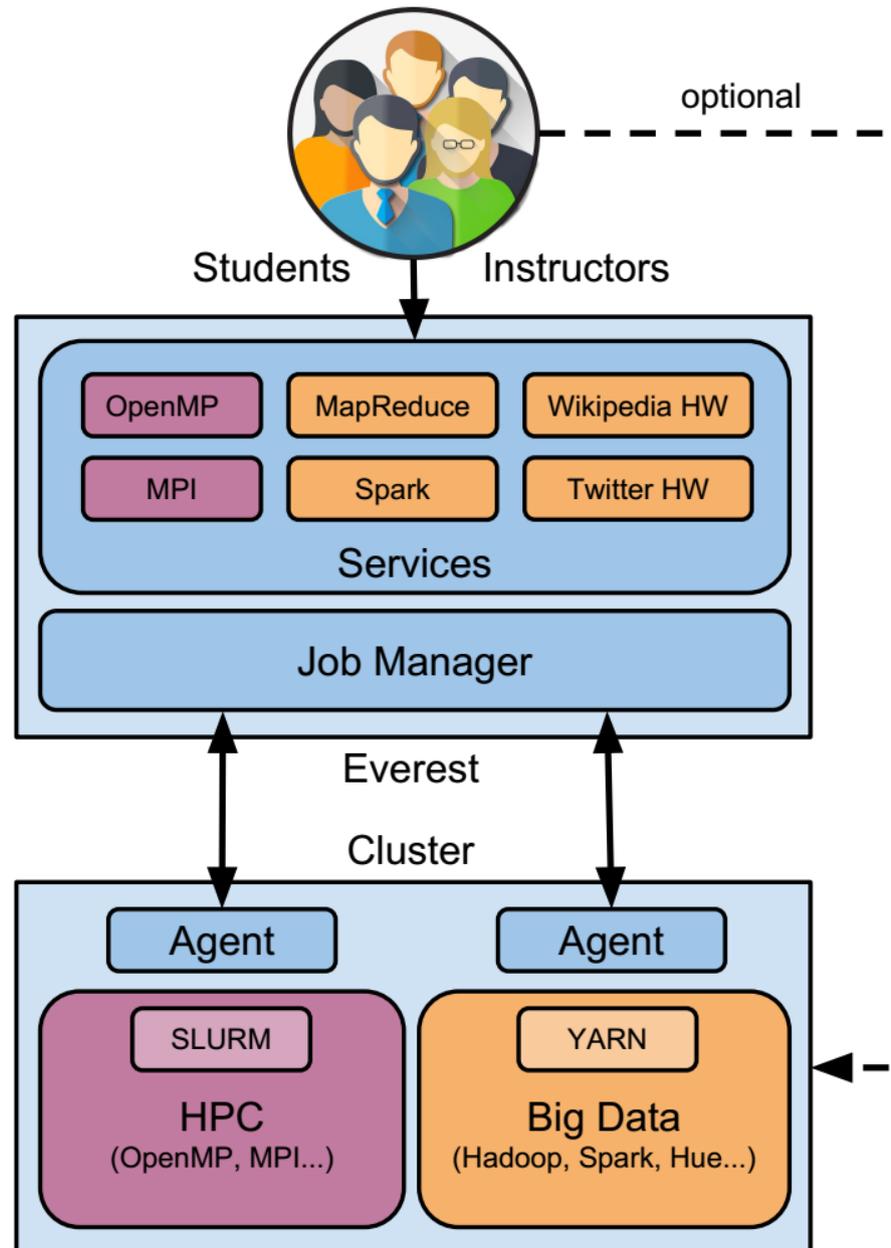
Введение

- Феномен Big Data
 - Новые модели программирования (MapReduce, Spark RDD)
 - Технологии распределенной обработки данных (Hadoop, Spark)
- Сложности включения технологий Big Data в традиционный курс по параллельным вычислениям
 - Заметные отличия от традиционных технологий параллельного программирования, другие языки и модели программирования
 - Использование специальной вычислительной инфраструктуры (Hadoop-кластер)
 - Практические занятия требуют освоения специфических интерфейсов командной строки

Опыт преподавания

- Курс «Параллельные и распределенные вычисления»
 - Школа анализа данных (с 2008 года)
 - Факультет компьютерных наук ВШЭ (с 2015 год)
- Основные разделы
 - Многопоточное программирование (C++)
 - Параллельные вычисления (OpenMP, MPI)
 - Распределенная обработка данных (Hadoop MapReduce, Spark)
 - Распределенные вычисления (Python)

Инфраструктура



Преимущества сервисов

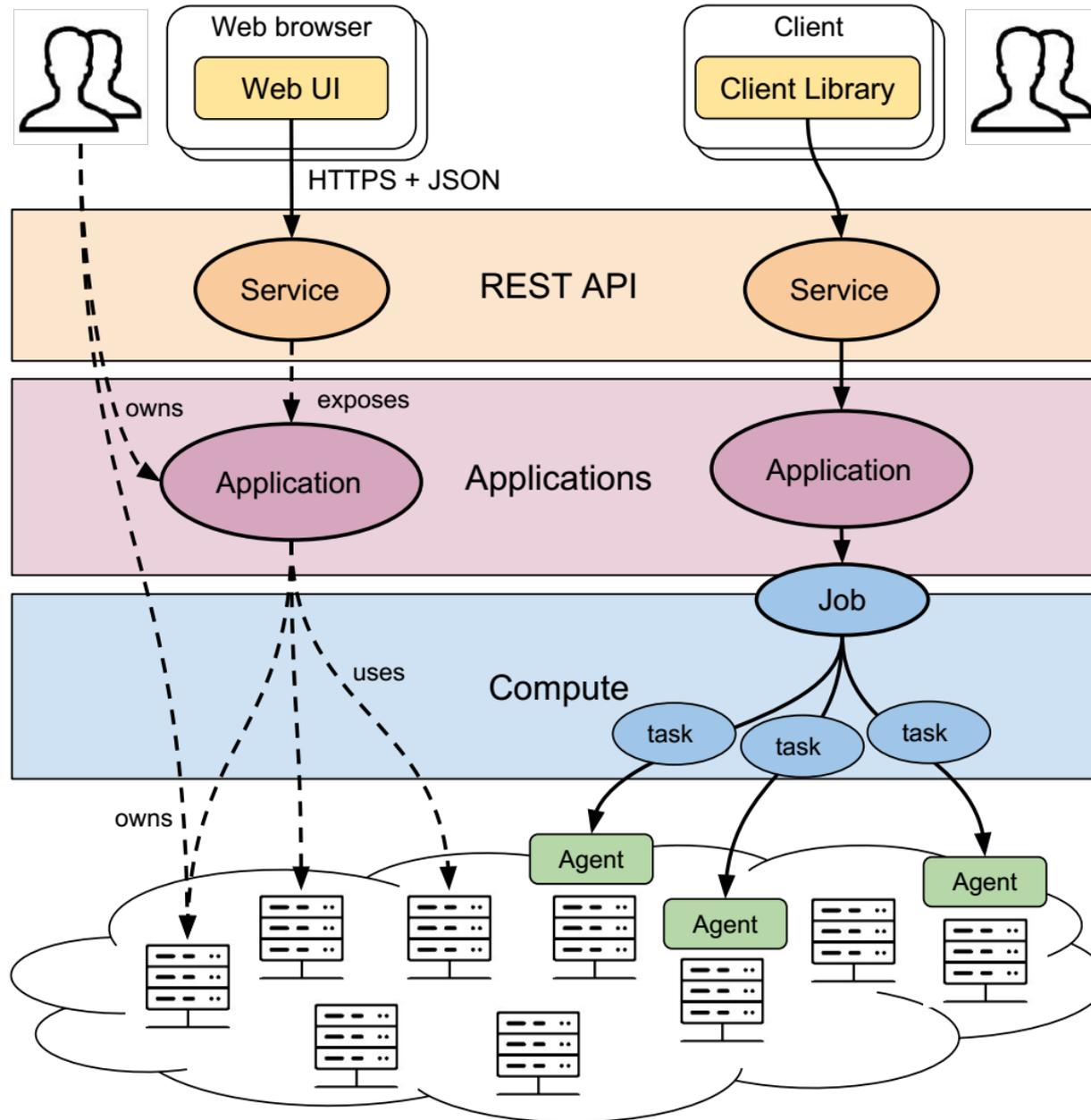
- Удобство использования в сравнении с интерфейсами командной строки
- Автоматизация рутинных действий (передача файлов, компиляция)
- Уменьшение нагрузки по администрированию пользователей
- Универсальные сервисы для запуска программ
 - помогают студентам быстро применять полученные знания
 - позволяют преподавателям включать в занятия больше упражнений и демонстраций
- Сервисы для тестирования решений заданий
 - позволяют студентам получить оперативный отклик в процессе решения задания
 - помогают преподавателям ускорить процесс оценки решений

Учебный кластер

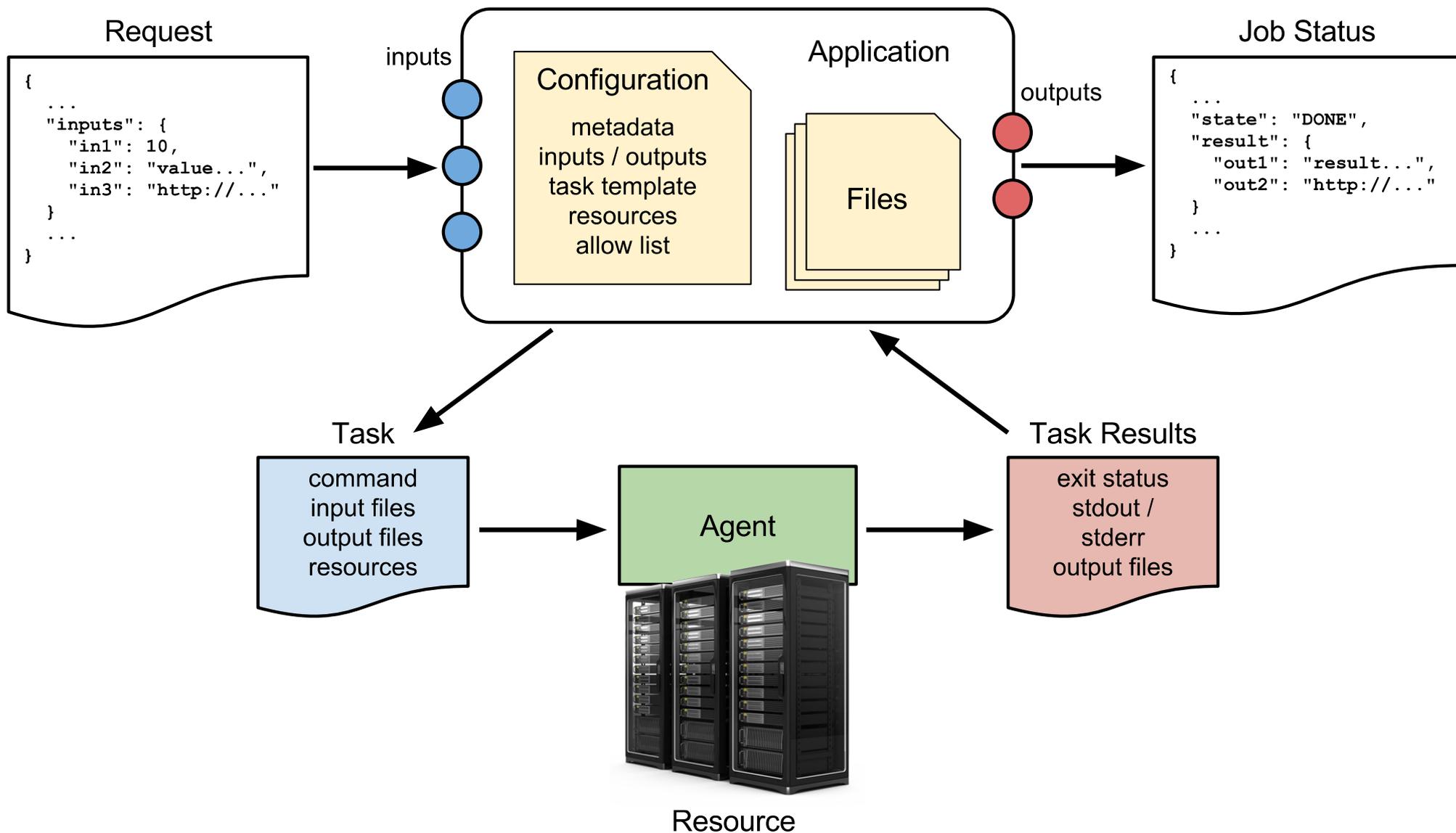
- Отличия технологий Big Data от HPC
 - Другие менеджеры ресурсов (YARN)
 - Динамическое выделение ресурсов в процессе выполнения программы
- Кластер разделен на две части
 - HPC: SLURM, NFS, OpenMPI
 - Hadoop: YARN, HDFS, MapReduce, Spark, Hive, HBase, Kafka
 - Размер каждой части настраивается в ручном режиме
- Лимиты потребления ресурсов
 - HPC: wall clock time
 - Hadoop: vcore-seconds

- Программная платформа, реализующая
 - Публикацию вычислительных приложений в виде веб-сервисов
 - Запуск приложений на внешних вычислительных ресурсах
 - Совместный доступ пользователей к приложениям и ресурсам
 - Композицию приложений
- Platform as a Service (PaaS)
 - Удаленный доступ через веб-браузер и интерфейс REST API
 - Один экземпляр платформы обслуживает много пользователей
- <http://everest.distcomp.org/>

Архитектура Everest



Приложение Everest (сервис)



Сервисы для технологий Big Data

- Сервисы для запуска MapReduce- and Spark-программ на учебном кластере
 - Два варианта сервиса для каждой технологии: Python и Java/Scala
 - Запуск заданий через менеджер ресурсов Hadoop YARN
 - Проксирование веб-интерфейсов Hadoop/Spark
 - Измерение объема потребленных ресурсов в ядро-секундах
- Сервисы для тестирования решений заданий
 - Построение инвертированного индекса Википедии (MapReduce)
 - Анализ графа пользователей Твиттера (Spark)
 - Только проверяют корректность полученных результатов

Hadoop MapReduce (Streaming)

About Parameters **Submit Job** Discussion

Job Name

Input path
Path to the file or directory in HDFS you want to use as the input data for the job.

Output path
Path to the directory in HDFS where you want to save the output of the job.

Mapper command
Command to use as mapper, e.g. "mapper.py"

Reducer command
Command to use as reducer, e.g. "reducer.py"

Combiner command
Optional command to use as combiner, e.g. "combiner.py"

Number of reducers
Specify the number of reduce tasks you want to use (maximum is 50). Specify zero if you do not want to run any reducer tasks.

Required files

Specify files implementing mapper, reducer, or combiner.

Options
Additional options to pass to Hadoop.

Word Count on Wikipedia

Job Info	Inputs	Outputs	Share	Tasks
Application	Hadoop MapReduce (Streaming)			
State	RUNNING			
Submitted	20 Aug 2016 20:43:55			
Finished				
Info	Hadoop job is running, status: http://mapreduce.distcomp.org/proxy/application_1470909240314_0012 (CoreSeconds: 3632)			
Log	view			



MapReduce Job
job_1400191378494_0002

Job Overview

Job Name: scopedTarget.wordcountJob
User Name: hillert
Queue: default
State: SUCCEEDED
Uberized: false
Started: Thu May 15 22:09:22 EDT 2014
Finished: Thu May 15 22:09:38 EDT 2014
Elapsed: 15sec

Diagnostics:

Average Map Time: 5sec
Average Reduce Time: 0sec
Average Shuffle Time: 4sec
Average Merge Time: 0sec

ApplicationMaster		Start Time	Node	Logs
1	Attempt Number	Thu May 15 22:09:18 EDT 2014	10.0.1.4:8042	logs

Task Type	Total	Complete	
Map	1	1	
Reduce	1	1	
Attempt Type	Failed	Killed	Successful
Maps	0	0	1
Reduces	0	0	1

Job Info	Inputs	Outputs	Share	Tasks
Application	Hadoop MapReduce (Streaming)			
State	DONE			
Submitted	20 Aug 2016 20:43:55			
Finished	20 Aug 2016 20:47:41			
Info	Hadoop job is completed, history: http://mapreduce.distcomp.org/jobhistory/job/job_1470909240314_0012 (CoreSeconds: 16603)			
Log	view			

PySpark

[About](#)[Parameters](#)[Submit Job](#)[Presets](#)[Discussion](#)**Job Name****Program**[+ Add file...](#)

Python program using PySpark.

Arguments

Program arguments.

Executors

Number of Spark executors (maximum 40).

Executor cores

Number of cores to use on each executor (maximum 4).

Executor memory

Amount of memory in GB to use for the executor process (maximum 18 GB, float values are supported).

Min executors

The minimum ratio of registered executors to wait for before starting computations. Specified as a double between 0.0 and 1.0.

Email Notification Send me email when the job completes

Apache HUE

The screenshot shows the Apache HUE File Browser interface. The top navigation bar includes 'HUE', 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and the user 'jennykim'. The main content area is titled 'File Browser' and contains a search bar, action buttons like 'Actions' and 'Move to trash', and 'Upload' and 'New' buttons. The breadcrumb path is 'Home / user / jennykim / 2015 / 02 / 06 / 00'. Below this is a table listing files and folders:

<input type="checkbox"/>	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	↑		jennykim	jennykim	drwxr-xr-x	July 28, 2015 02:02 PM
<input type="checkbox"/>	.		jennykim	jennykim	drwxr-xr-x	July 28, 2015 02:21 PM
<input type="checkbox"/>	2015-02-06-00.txt	35 bytes	jennykim	jennykim	-rw-r--r--	July 28, 2015 02:21 PM
<input type="checkbox"/>	2015-02-06-01.txt	35 bytes	jennykim	jennykim	-rw-r--r--	July 28, 2015 02:21 PM
<input type="checkbox"/>	2015-02-06-03.txt	35 bytes	jennykim	jennykim	-rw-r--r--	July 28, 2015 02:21 PM
<input type="checkbox"/>	2015-02-06-04.txt	35 bytes	jennykim	jennykim	-rw-r--r--	July 28, 2015 02:21 PM

At the bottom, there is a pagination control showing 'Show 45 of 4 items' and 'Page 1 of 1' with navigation arrows.

- About
- Parameters
- Submit Job**
- Presets
- Discussion

Annotation	Задача на распределенную обработку данных с Hadoop MapReduce
Keywords	cdc big data MapReduce Hadoop
Owner	sol
State	READY
Times Used	429
Rating	★ 3
Last Updated	24 Jun 2017 14:59:17
Latest Version	0.1.0
ID	5354d1de3d0000a72f290835
Service URI	https://everest.distcomp.org:443/apps/535

- About
- Parameters
- Submit Job**
- Presets
- Discussion

Job Name

English Wikipedia Index
путь к папке с построенным индексом английской Википедии в HDFS

Russian Wikipedia Index
путь к папке с построенным индексом русской Википедии в HDFS

Постановка задачи

Построить инвертированные индексы для русской и английской Википедий.

Индекс должен состоять из строк вида `word \t docid1:tfidf1 \t docid2:tfidf2 ...`, где `word` - слово, `docid` - идентификатор статьи, `tfidf` - значение меры TD-IDF для пары слово-статья, `\t` - символ табуляции (пробелов вокруг него быть не должно).

Для каждого слова должно быть включено 20 наиболее релевантных статей в порядке убывания их TF-IDF. При равных значениях TF-IDF статьи должны располагаться в порядке возрастания `docid`.

При построении индекса следует игнорировать 20 наиболее часто встречающихся в текстах статей слов, используя предварительно вычисленный список данных слов.

Требования к парсингу слов минимальные. Обязательно только отбрасывание знаков пунктуации и приведение к нижнему регистру. Для простоты в тексте русской Википедии можно вообще оставлять только кириллические символы, а в английской - латинские. Приведение слов к нормальной форме не требуется.

Дополнительно требуется минимизировать время построения индекса и обеспечить масштабируемость реализации, то есть возможность эффективно обрабатывать данные большего объема и использовать большее число машин.

Заключение

- Рассмотрен опыт включения технологий Big Data в курс по параллельным и распределенным вычислениям
 - Организация вычислительной и программной инфраструктуры для поддержки практических занятий
 - Реализация сервисов для запуска программ и тестирования решений заданий на учебном кластере
- Дальнейшие планы
 - Совершенствование описанной инфраструктуры
 - Публикация готовых шаблонов сервисов для использования в других курсах