MapReduce Fundamentals: Demo

Семинар курса «Управление разно-структурированными большими данными»

http://synthesis.ipi.ac.ru/synthesis/student/BigData/seminar-hadoop/hadoop2014

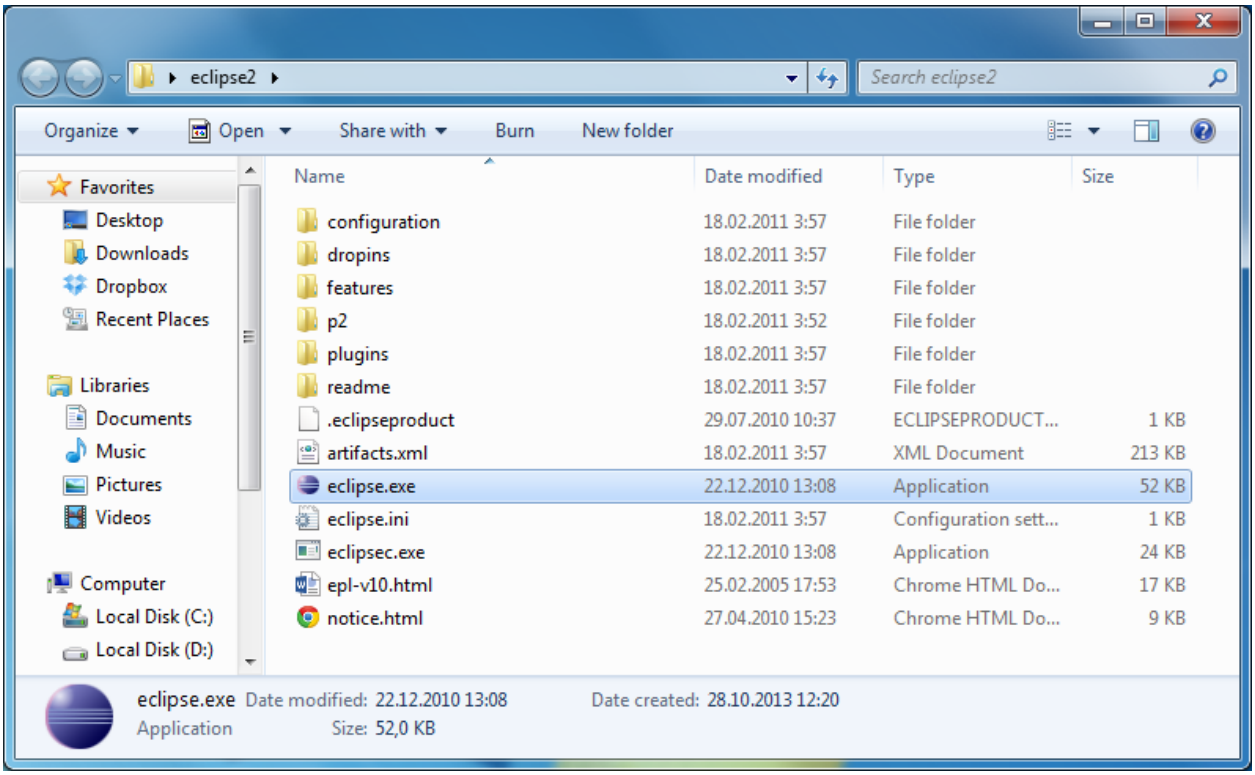
alexey.vovchenko@gmail.com

Начало работы

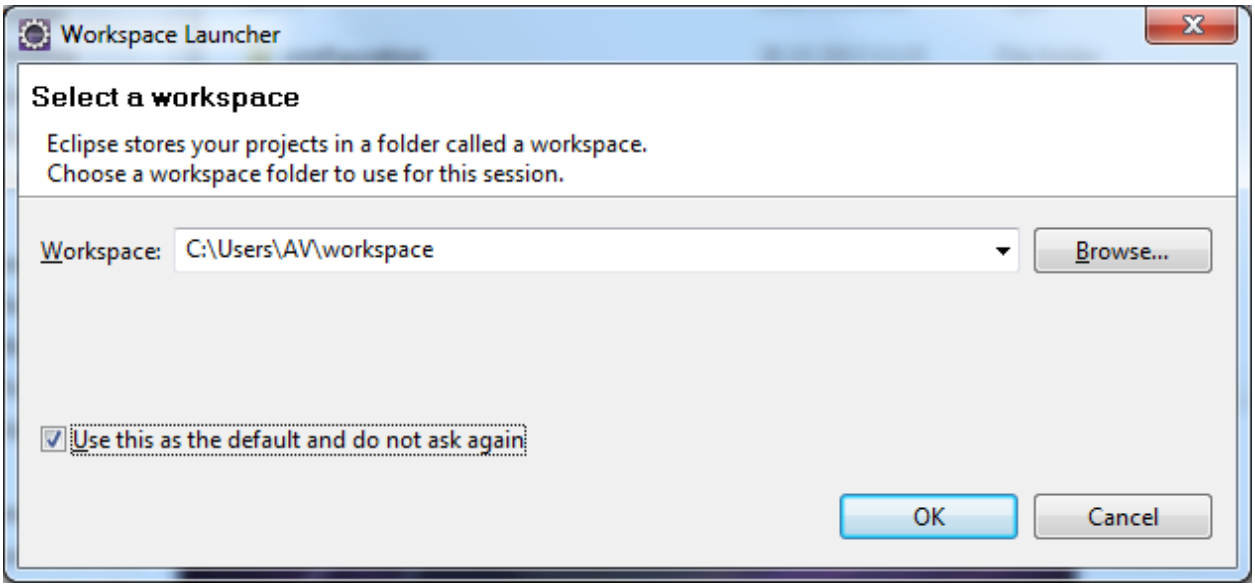
Для доступа к серверу IBM BigInsights необходимо выполнить инструкции из: «server_access.pdf». Если у Вас нет этого файла, необходимо написать мне на почту.

Подготовка среды разработки

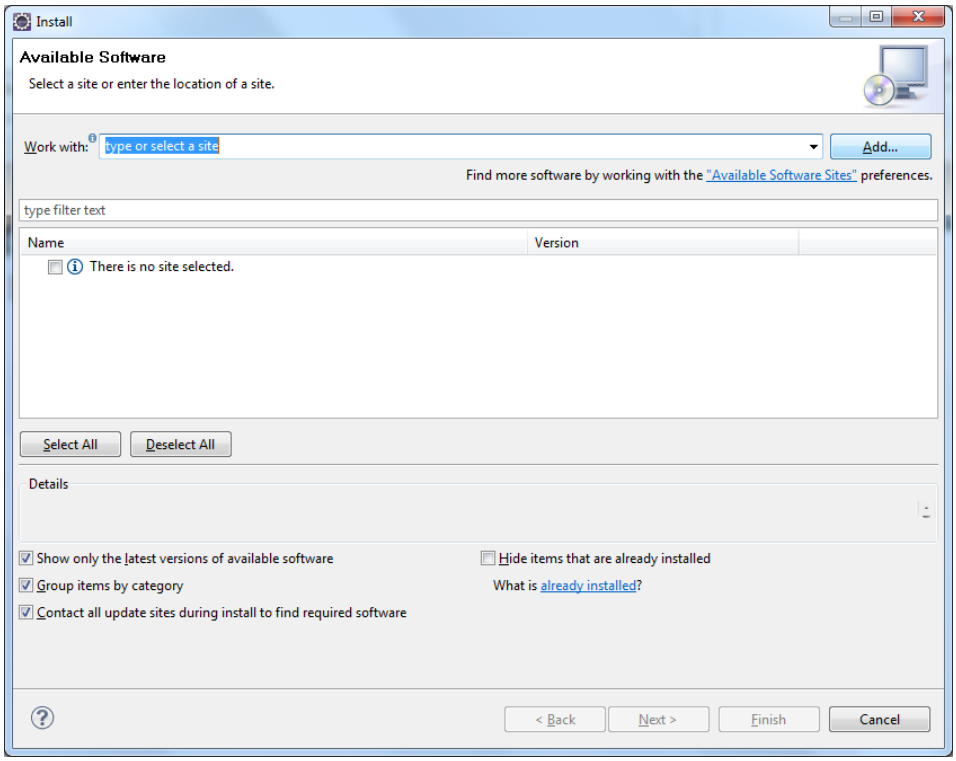
1. Скачать и установить Java 7 (если не установлен, на Java 8 работать не будет) http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html
2. Скачать Eclipse версии 4.2 https://www.eclipse.org/downloads/packages/release/juno/sr2
3. Скачать инструменты разработки IBM BigInsights http://83.149.245.126:8080/updatesite/repository
4. Запускаем Eclipse версии 4.2



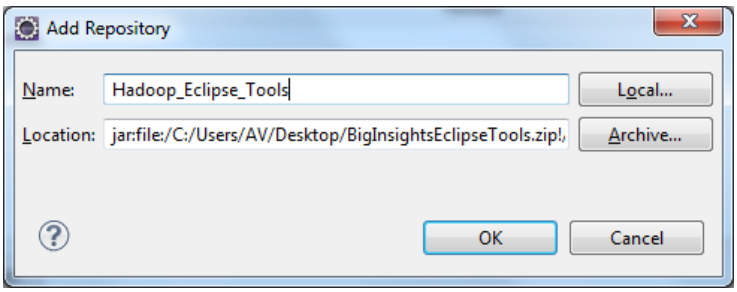
5. Указываем папку где будут хранится наши проекты



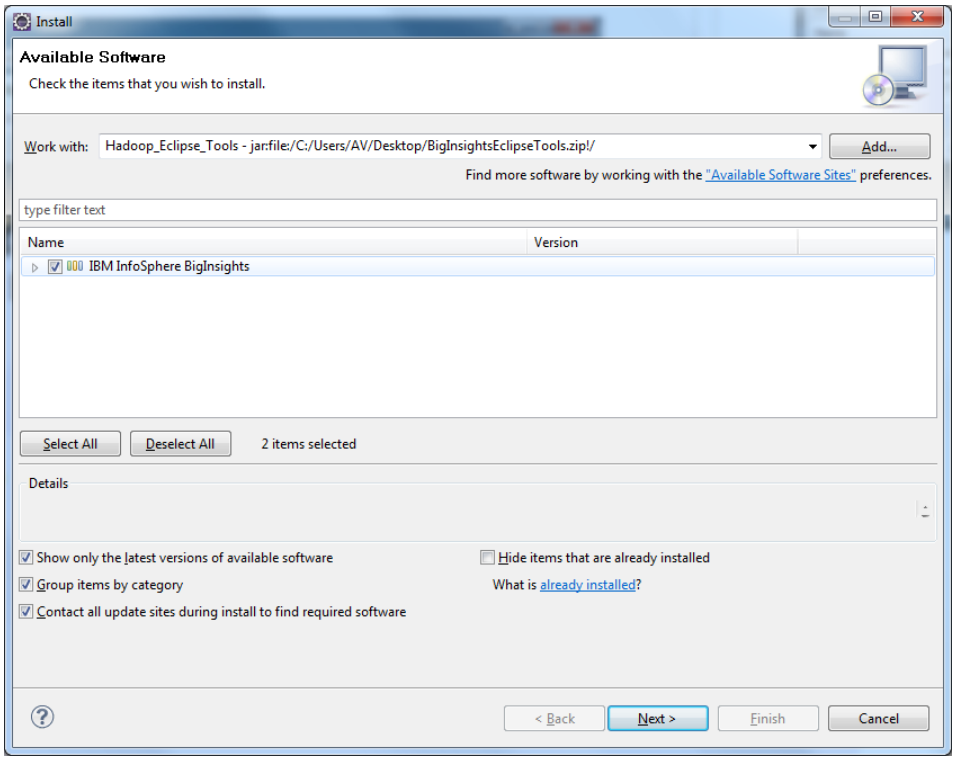
6. Открываем диалог установки дополнений: Help → Install New Software…



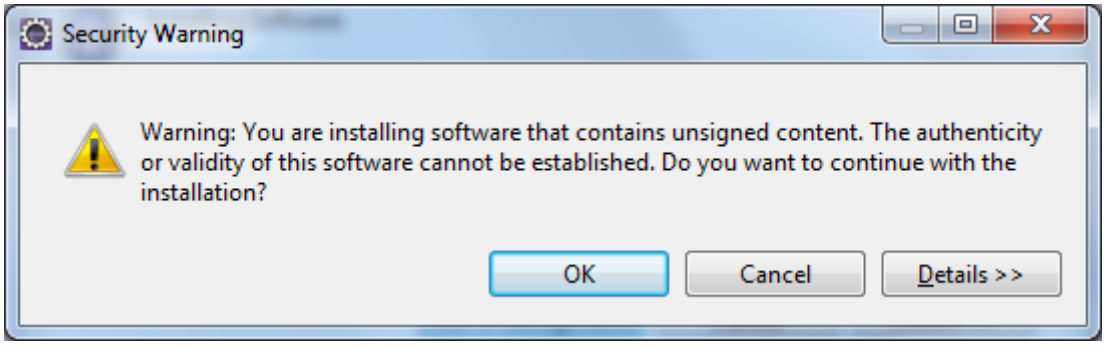
7. Нажимаем кнопку добавить (Add…) В появившемся окне выбираем для установки архив, содержащий BigInsights Eclipse Tools (BigInsightsEclipseTools.zip)



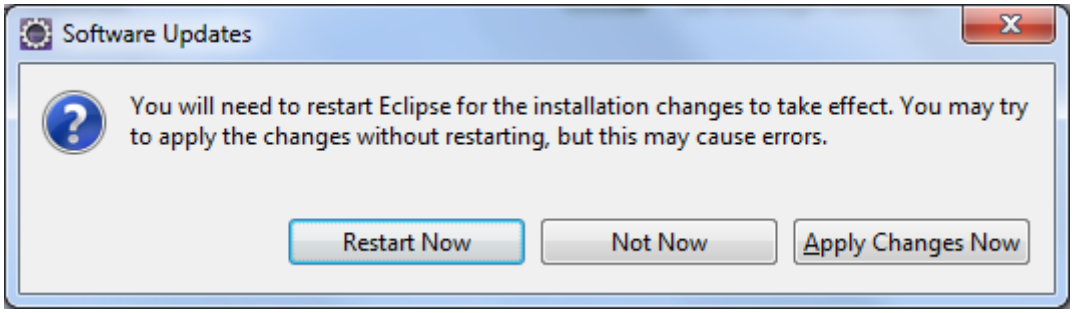
8. Выбираемся для установки IBM InfoSphere BigInsights и нажимаем Next



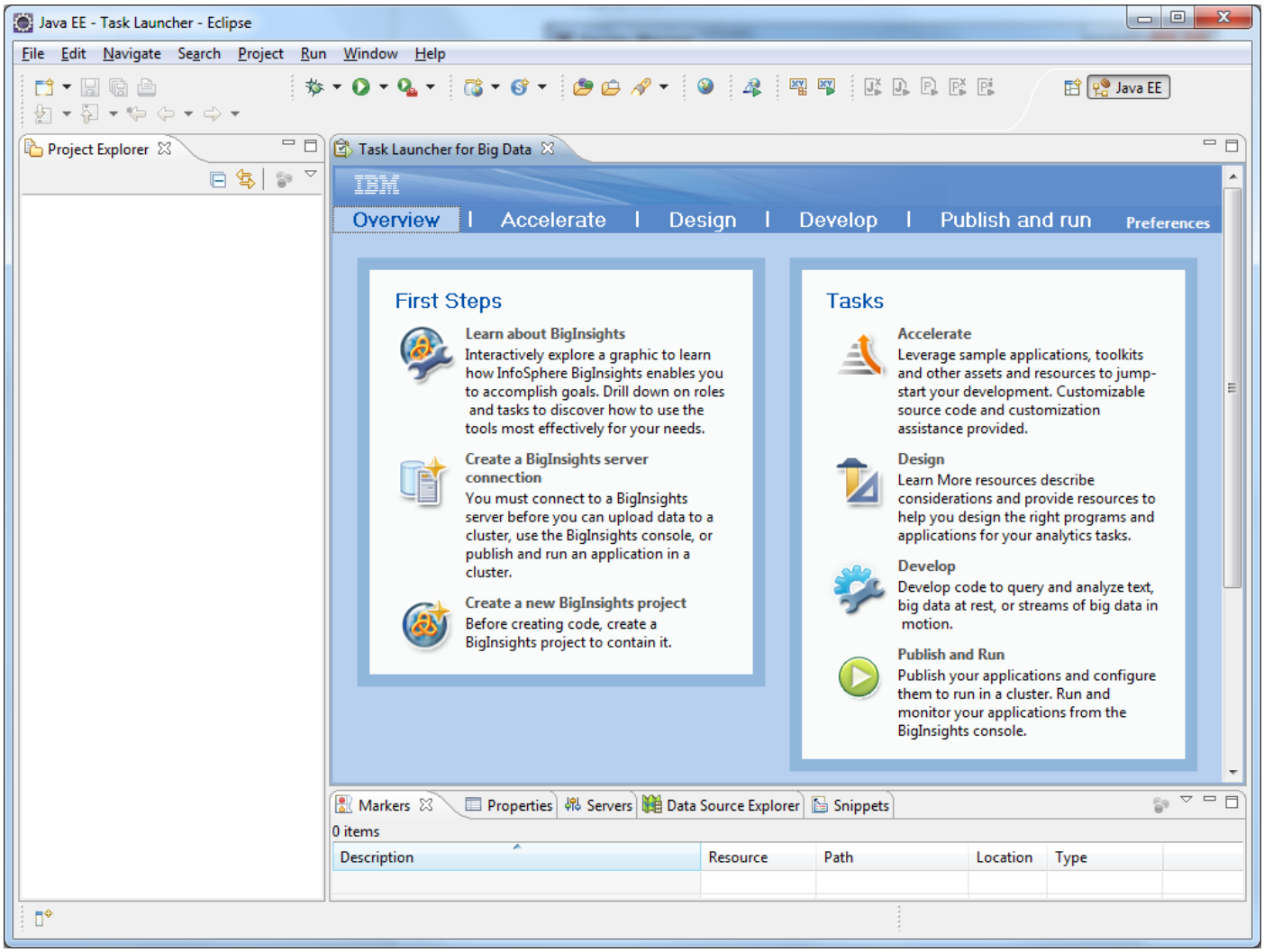
9. Устанавливаем выбранный плагин, соглашаясь со всеми предложениями мастера по установке и всеми предупреждениями.



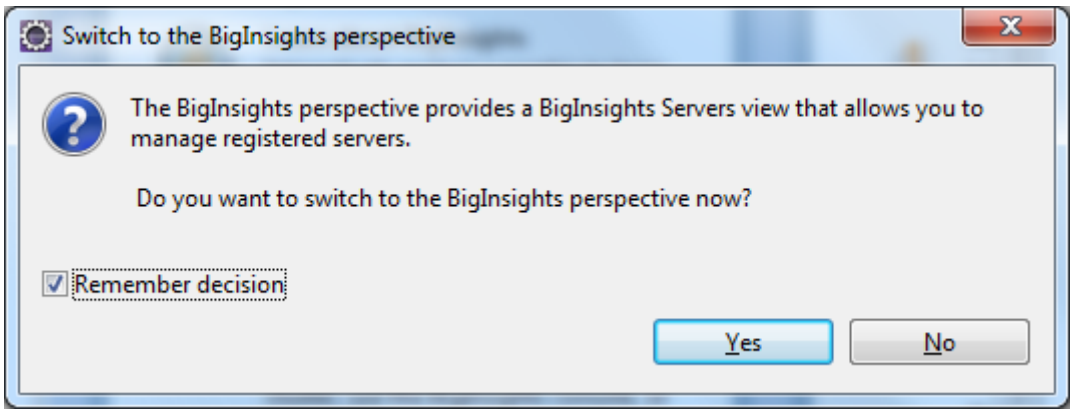
10. После установки перезагружаем Eclipse



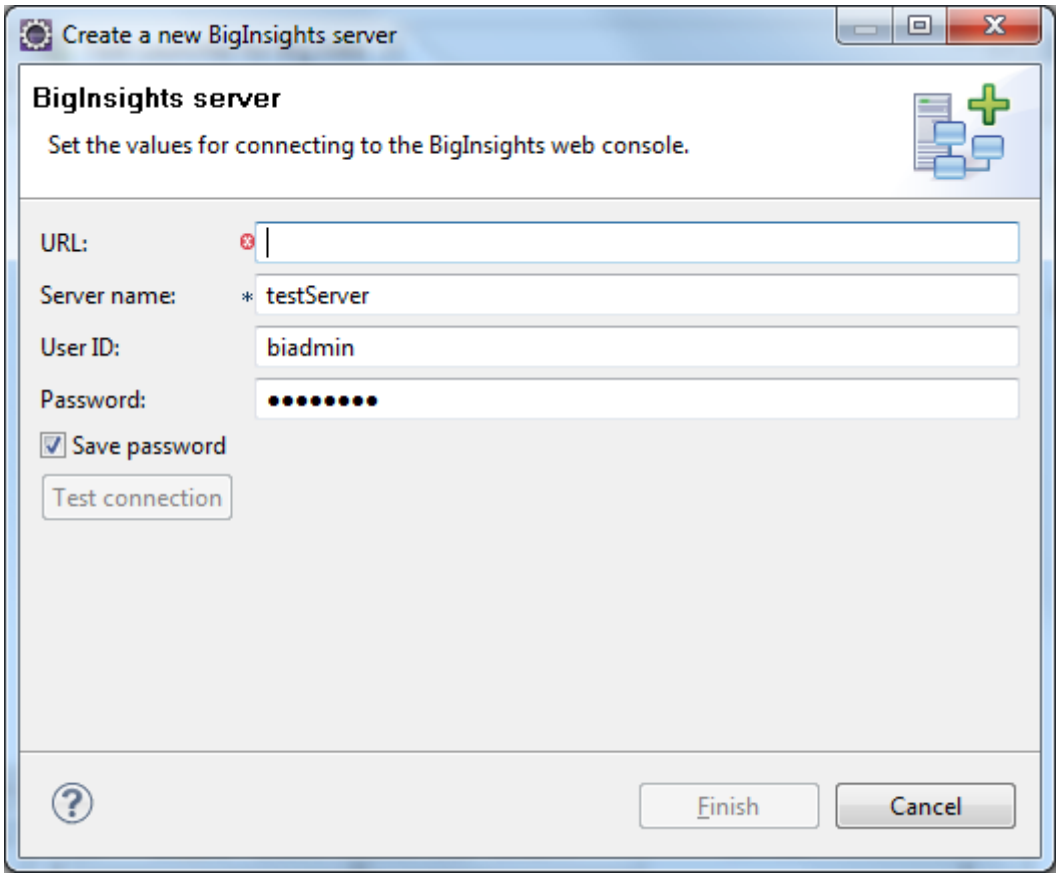
11.В открывшимся окне нажимаем «Create a BigInsights server connection»



12.Соглашаемся изменить перспективу



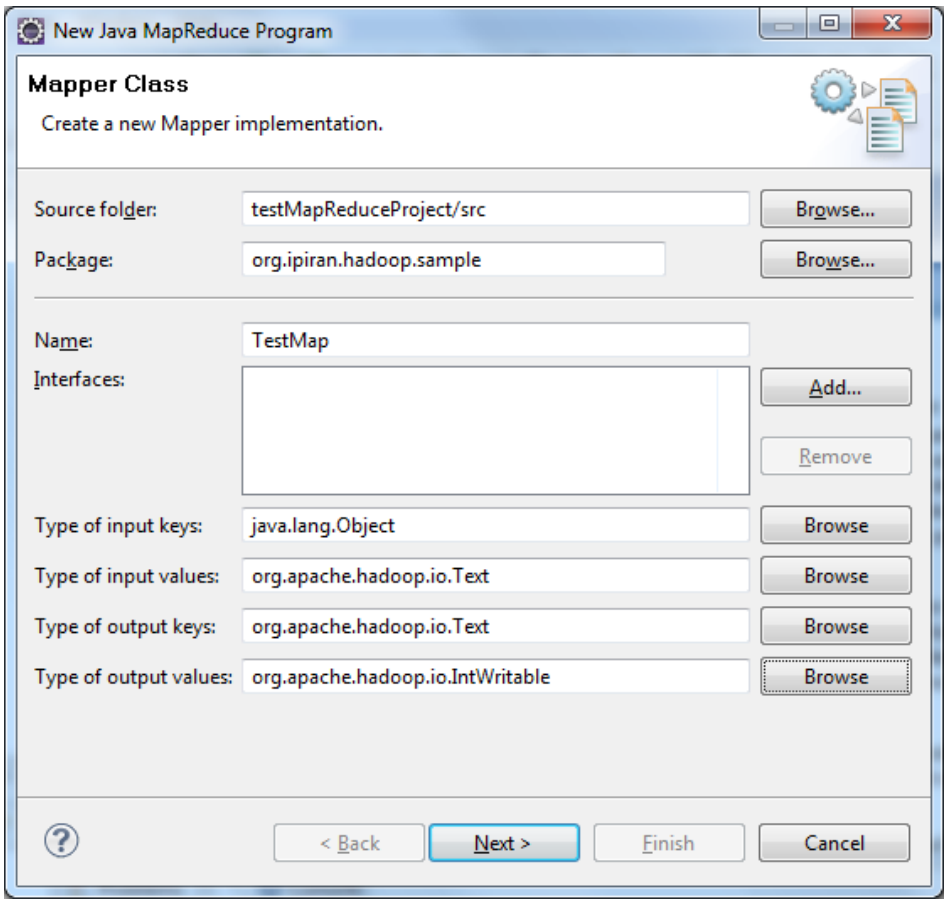
13. Вводим URL сервера (например http://1.1.1.1:8080), имя сервера, логин и пароль



Создаем собственное MapReduce приложение по подсчету числа слов в тексте

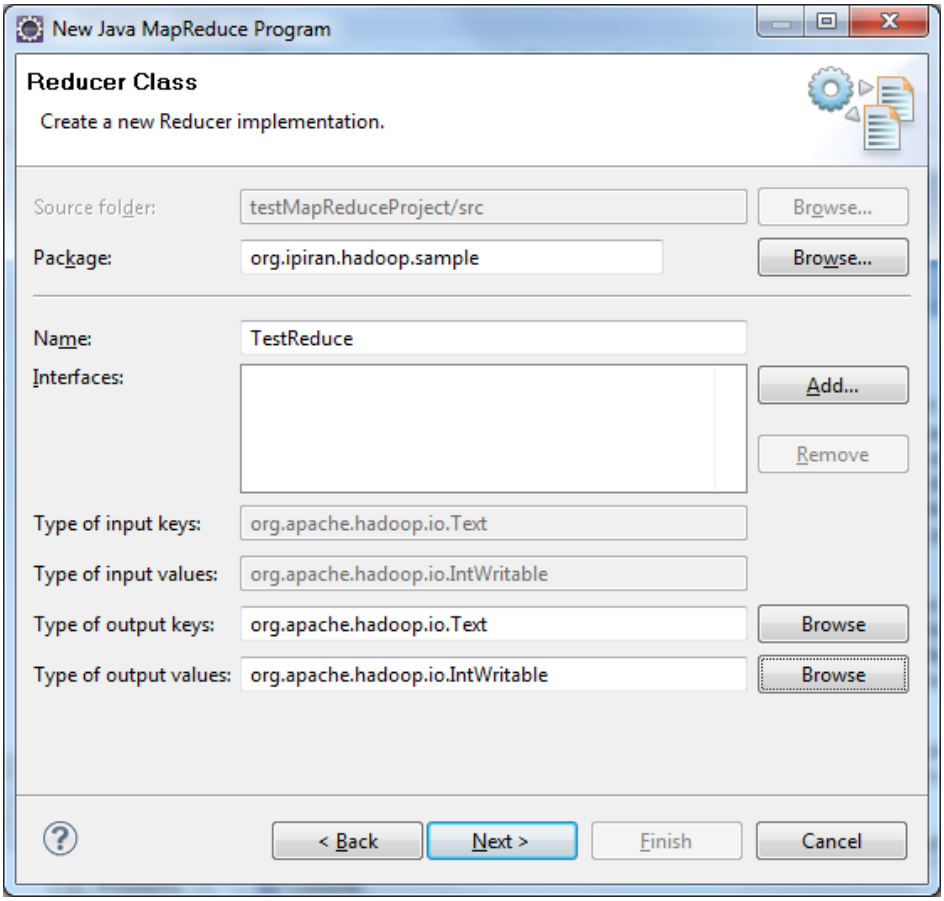
1. Создаем проект File → New → BigInsights Project
2. После того как проект создан, выделяем проект и нажимаем File → New → Java MapReduce Program
3. Заполняем параметры для Mapper класса

Package	org.ipiran.hadoop.sample
Name	TestMap
Type of input keys	java.lang.Object
Type of Input values	org.apache.hadoop.io.Text
Type of output keys	org.apache.hadoop.io.Text
Type of output values	org.apache.hadoop.io.IntWritable



4. Заполняем параметры для Redcuer класса

Name	TestReduce
Type of output keys	org.apache.hadoop.io.Text
Type of output values	org.apache.hadoop.io.IntWritable



5. Заполняем параметры для Main класса

Package	org.ipiran.hadoop.sample
Name	TestMapReduce

6. В файле TestMapReduce.java исправляем две строки

```
// TODO: Update the input path for the location of the inputs of the map-reduce…
FileInputFormat.addInputPath(job, new Path("[input path]"));
// TODO: Update the output path for the output directory of the map-reduce job.
FileOutputFormat.setOutputPath(job, new Path("[output path]"));
```

На

```
// TODO: Update the input path for the location of the inputs of the map-reduce…
FileInputFormat.addInputPath(job, new Path(programArgs[0]));
// TODO: Update the output path for the output directory of the map-reduce job.
FileOutputFormat.setOutputPath(job, new Path(programArgs[1]));
```

7. Пишем код для Mapper класса

```
package org.ipiran.hadoop.sample;

import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class TestMap extends Mapper<Object, Text, Text, IntWritable> {
      private final static IntWritable ONE = new IntWritable(1);

      @Override
      public void map(Object key, Text value, Context context)
                  throws IOException, InterruptedException {
            StringTokenizer tokens = new StringTokenizer(value.toString());
            while (tokens.hasMoreTokens()) {
```

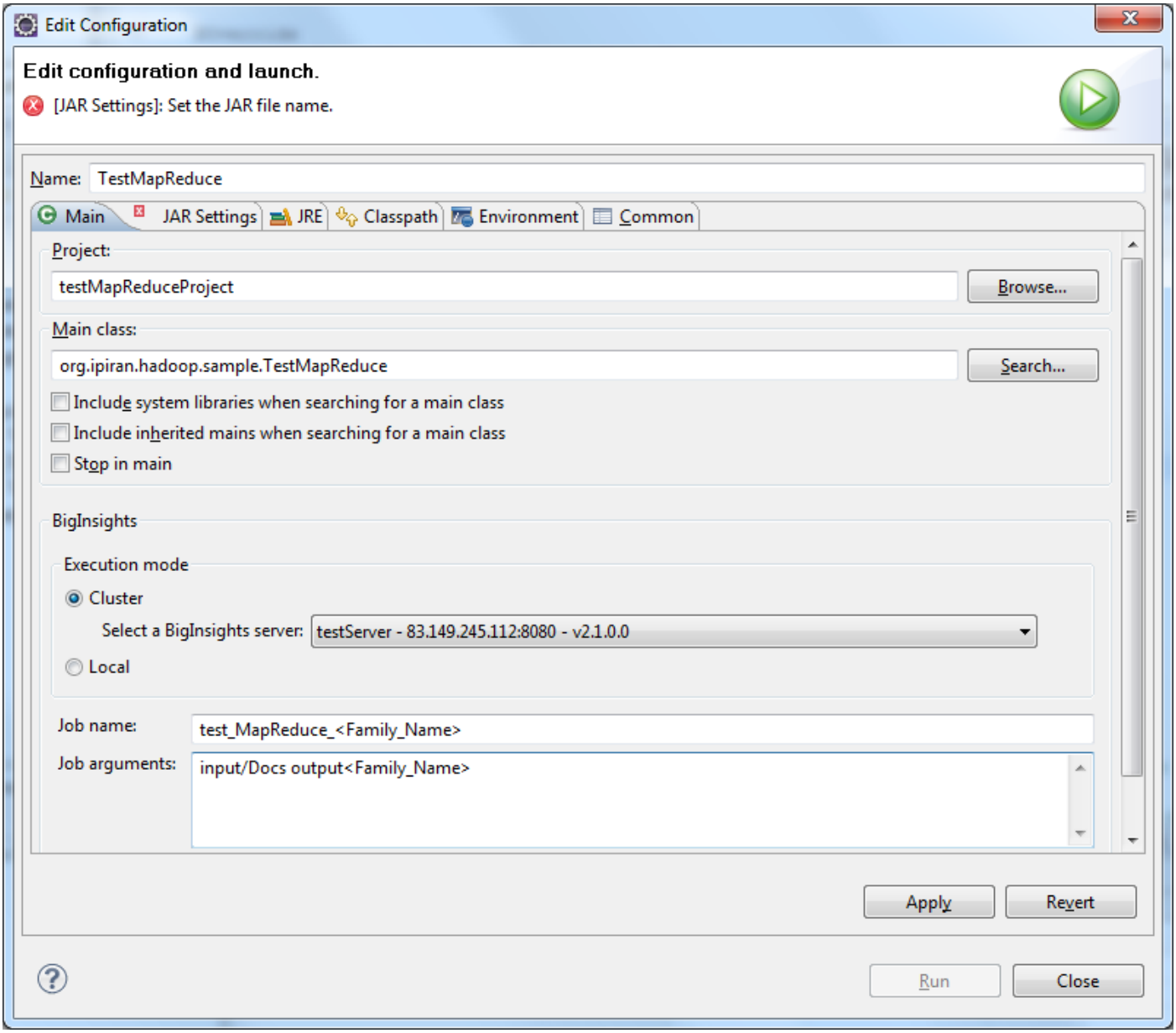
```
                    Text word = new Text();
                    word.set(tokens.nextToken());
                    context.write(word, ONE);
                }
        }
}
```

8. Пишем код для Reducer класса

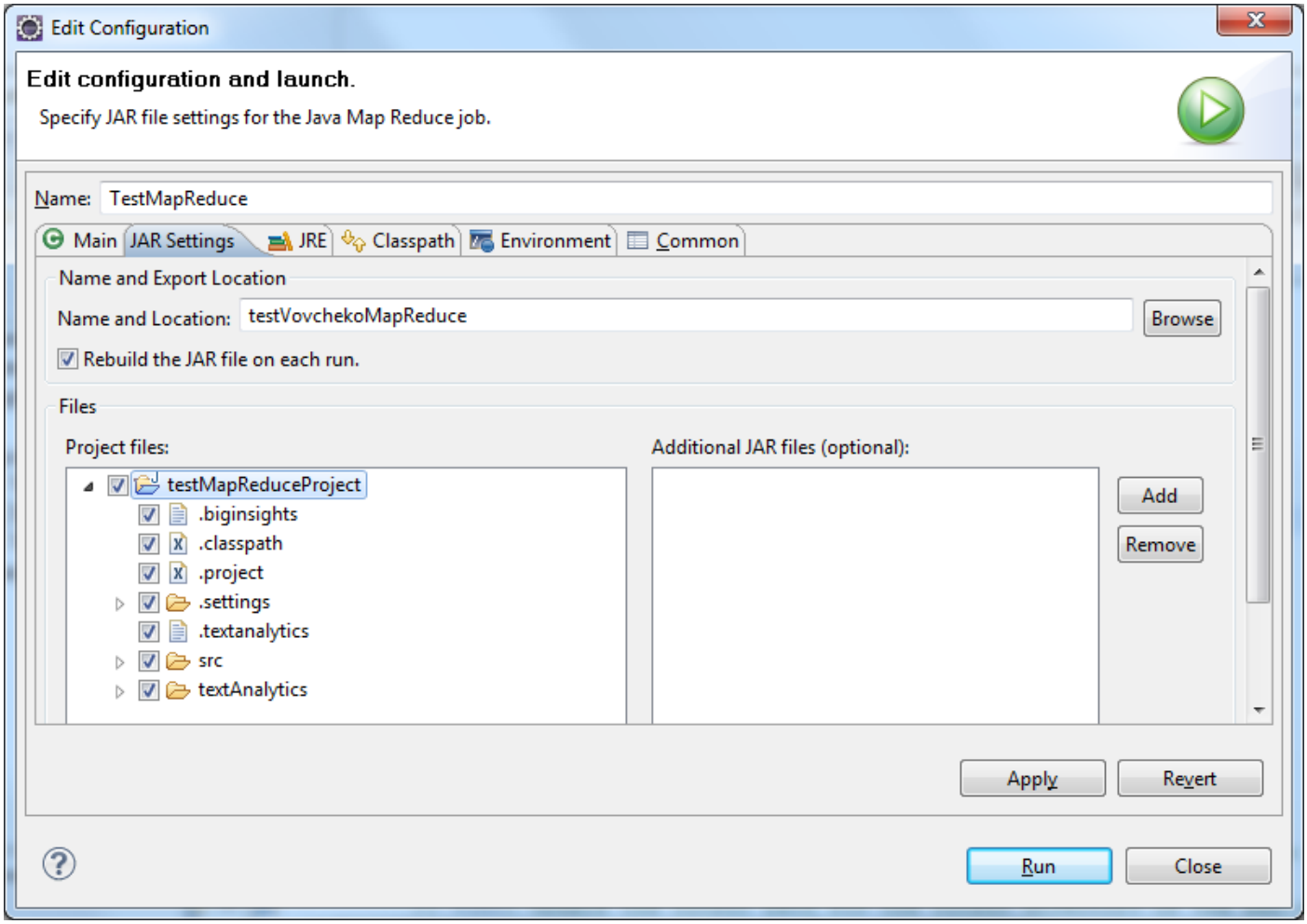
```
package org.ipiran.hadoop.sample;

import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class TestReduce extends Reducer<Text, IntWritable, Text, IntWritable> {

      public void reduce(Text key, Iterable<IntWritable> values, Context context)
                  throws IOException, InterruptedException {
            IntWritable result = new IntWritable();
            int sum = 0;
            for (IntWritable value: values) {
                  sum += value.get();
            }
            result.set(sum);
            context.write(key, result);
      }
}
```

9. Открываем файл TestMapReduce.java и выбираем из меню: Run → Run as → Java MapReduce
10. Заполняем параметры конфигурации Job Name и аргументы (путь к входным данным и путь для результата). **Важно!** Вместо <Family_Name> нужно подставить любой текст, например свою фамилию. Это нужно для того, чтобы имена разных студентов не пересекались. Это же имя нужно использовать в пункте 15 и 16.

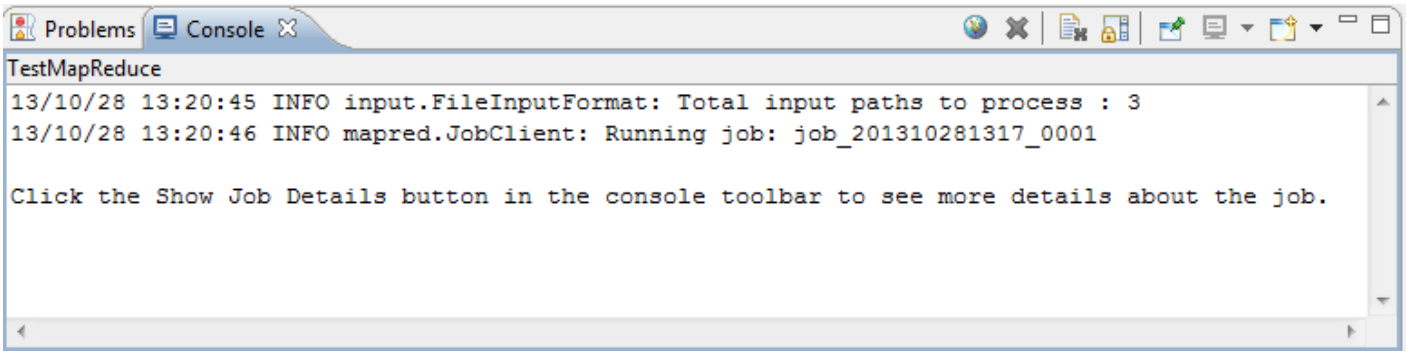
Job name	Test_MapReduce_<Family_Name>
Job arguments	<Family_Name>/hadoop_lab/texts <Family_Name>/mr_out1


Edit Configuration
Edit configuration and launch.
[JAR Settings]: Set the JAR file name.
Name: TestMapReduce
Main
JAR Settings
JRE
Classpath
Environment
Common
Project:
testMapReduceProject
Browse...
Main class:
org.ipiran.hadoop.sample.TestMapReduce
Search...
Include system libraries when searching for a main class
Include inherited mains when searching for a main class
Stop in main
BigInsights
Execution mode
Cluster
Select a BigInsights server:
testServer - 83.149.245.112:8080 - v2.1.0.0
Local
Job name:
test_MapReduce_<Family_Name>
Job arguments:
input/Docs output<Family_Name>
Apply
Revert
Run
Close

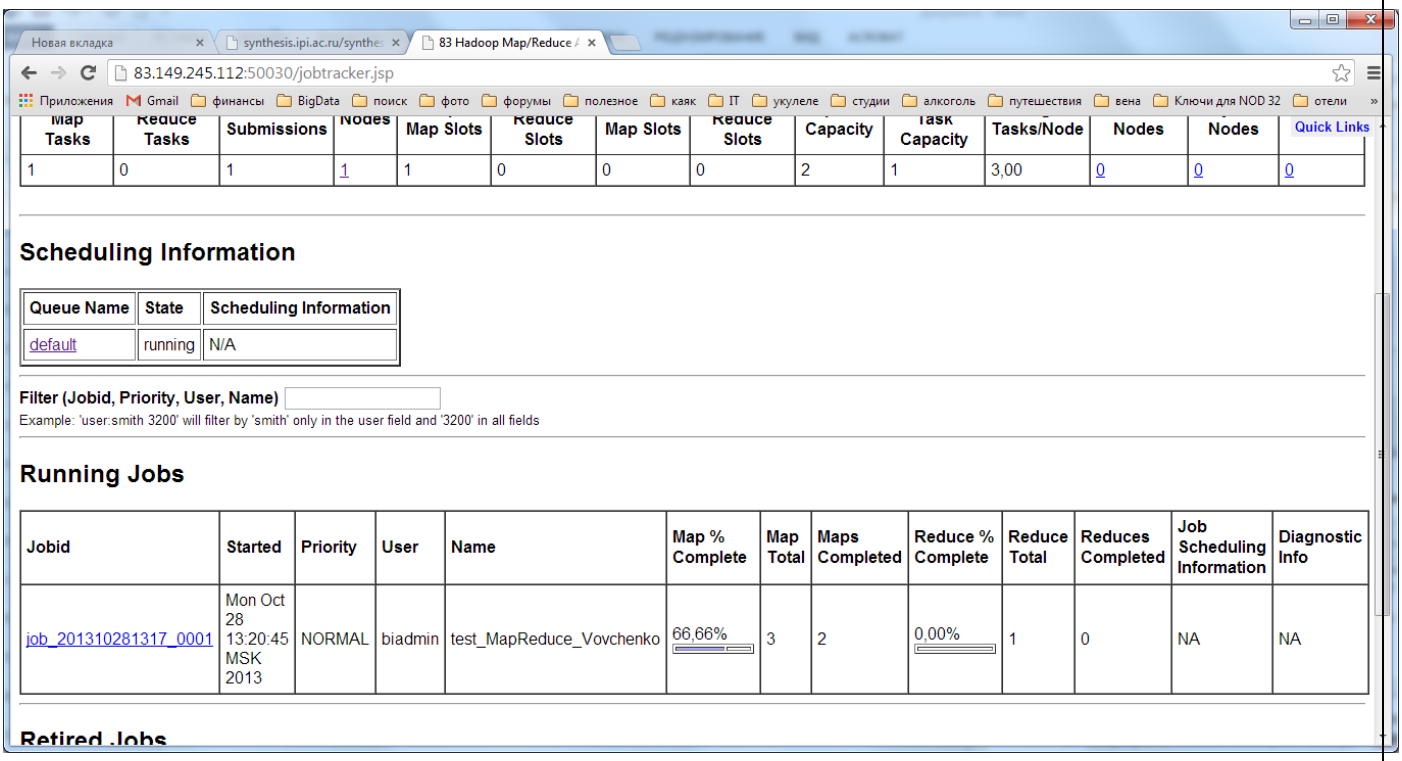
11. На закладке Jar Settings заполняем имя jar файла и отмечаем опцию «Rebuild the Jar file on each run»



12. Нажимаем Run
13. В консоли отображается информация о запущенном задании



14. Зайдя на страницу Hadoop Map/Reduce Administration http://83.149.245.126:50030/ , можно проверить статус собственного приложения



83.149.245.112:50030/jobtracker.jsp

Map Tasks	Reduce Tasks	Submissions	Nodes	Map Slots	Reduce Slots	Map Slots	Reduce Slots	Capacity	Task Capacity	Tasks/Node	Nodes	Nodes	Quick Links
1	0	1	1	1	0	0	0	2	1	3,00	0	0	0

Scheduling Information

Queue Name	State	Scheduling Information
default	running	N/A

Filter (Jobid, Priority, User, Name)

Example: 'user:smith 3200' will filter by 'smith' only in the user field and '3200' in all fields

Running Jobs

Jobid	Started	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total	Reduces Completed	Job Scheduling Information	Diagnostic Info
job_201310281317_0001	Mon Oct 28 13:20:45 MSK 2013	NORMAL	biadmin	test_MapReduce_Vovchenko	66,66%	3	2	0,00%	1	0	NA	NA

Retired Jobs

15. Содержимое папки с результатом можно посмотреть командой:
    hadoop fs -ls <Family_Name>/mr_out1
16. Сам результат можно посмотреть командой:
    hadoop fs -cat <Family_Name>/mr_out1/*00