Universidad
de La Laguna

ESCUELA TÉCNICA SUPERIOR
INGENIERÍA INFORMÁTICA

Hadoop - MapReduce
Restaurants close to Tenerife's beaches

Juan José Labrador González

Escuela Superior de Ingeniería y Tecnología
Computación de Altas Prestaciones y Tecnologías Web
Máster en Ingeniería Informática
Universidad de La Laguna

01 de Junio de 2015

Goal

We want to know the closest restaurants of each beach of Tenerife in order to:

- Inform and recommend to the tourist that visit the island where they can take a meal.
- Give new gastronomic options to the Tenerife population.
- Promote new restaurants.

Solution

- We use **Hadoop-MapReduce** to solve this approach.
- The input files of beaches and restaurants come from **OpenData Canarias**, specifically from **SPET, Turismo de Tenerife S.A.** Dataset.
- It's possible to change the distance value (in km) for the search.

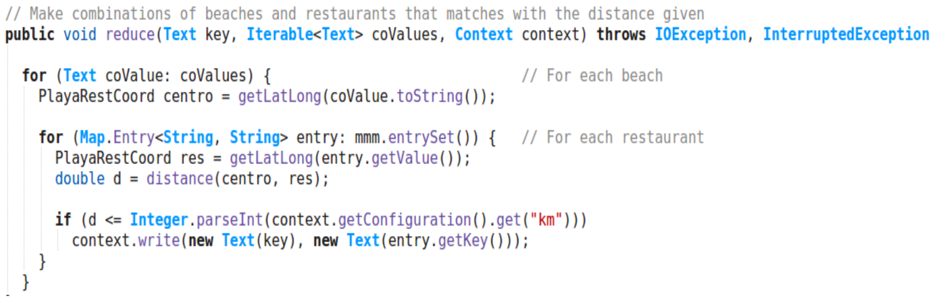
Mapper

```java
// Mapper
public static class PlayaRestMapper extends Mapper<Object, Text, Text, Text> {

  private static final String SEPARATOR = ",";

  public void map(Object key, Text value, Context context) throws IOException, InterruptedException {

    final String[] values = value.toString().split(SEPARATOR);

    final String playa = values[0].trim();     // Beach's name
    final String lat = values[2].trim();       // Latitude
    final String lon = values[3].trim();       // Longitude
    final String coord = lat + "," + lon;      // Coordinates

    context.write(new Text(playa), new Text(coord));
  }
}
```

Combiner I

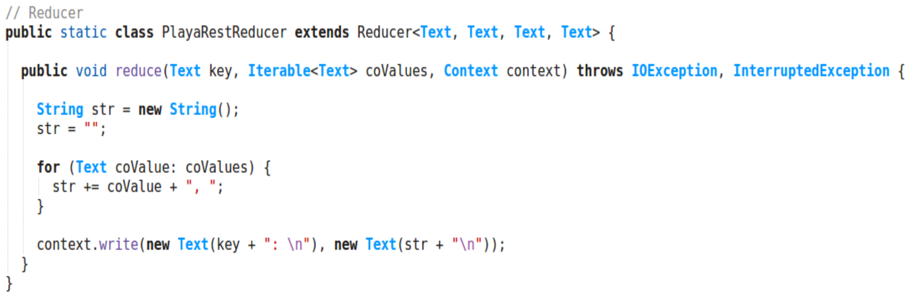
```
// Combiner
public static class PlayaRestCombiner extends Reducer<Text, Text, Text, Text> {

  // Hash that stores restaurants' coordinates
  Map<String, String> m = new HashMap<String, String> ();



BufferedReader reader = new BufferedReader(new FileReader("restauracion.csv"));

while ((strRead=reader.readLine() ) != null) {
  String splitarray[] = strRead.split(SEPARATOR);

  final String res = splitarray[0].trim();    // Restaurant's name
  final String lat = splitarray[2].trim();    // Latitude
  final String lon = splitarray[3].trim();    // Longitude
  final String coord = lat + "," + lon;       // Coordinates

  m.put(res, coord);
}
```

Combiner II



```java
// Make combinations of beaches and restaurants that matches with the distance given
public void reduce(Text key, Iterable<Text> coValues, Context context) throws IOException, InterruptedException

  for (Text coValue: coValues) {                              // For each beach
    PlayaRestCoord centro = getLatLong(coValue.toString());

    for (Map.Entry<String, String> entry: mmm.entrySet()) {   // For each restaurant
      PlayaRestCoord res = getLatLong(entry.getValue());
      double d = distance(centro, res);

      if (d <= Integer.parseInt(context.getConfiguration().get("km")))
        context.write(new Text(key), new Text(entry.getKey()));
    }
  }
```

Reducer



```
// Reducer
public static class PlayaRestReducer extends Reducer<Text, Text, Text, Text> {

  public void reduce(Text key, Iterable<Text> coValues, Context context) throws IOException, InterruptedException {

    String str = new String();
    str = "";

    for (Text coValue: coValues) {
      str += coValue + ", ";
    }

    context.write(new Text(key + ": \n"), new Text(str + "\n"));
  }
}
```

Running the example

bin/hadoop jar <.jar> <ClassName> <input dir> <output dir> **-files** <file.csv> **-D** <distance>

bin/hadoop jar pr.jar PlayaRest /user/hduser/input /user/hduser/output -**files** /opt/hadoop/jj/restauracion.csv -**D** 5

Problems

- Scope of classes' attributes (e.g: between Mappers).
- Need of customization of the context for each problem.
- Constraint of key-value data.

Bibliography

- "Latest stable documentation."
  https://hadoop.apache.org/docs/stable/.
- "Multiple input files."
  http://goo.gl/As5yoW.
- "Passing parameters to Mappers and Reducers."
  http://goo.gl/MDn7V8.
- "Basic MapReduce."
  http://www.adictosaltrabajo.com/tutoriales/mapreduce-basic/.

End

Thank you