From a NoSQL Data Source to a Business Intelligence Solution: An Experiment

Marco Costa
Universidade do Minho
Depart. de Sistemas de Informação
Guimarães
Portugal
A59742@alunos.uminho.pt

José Luís Pereira
Universidade do Minho
Depart. de Sistemas de Informação
Guimarães
Portugal
jlmp@dsi.uminho.pt

Abstract

We are living in the era of *Big Data*. A time which is characterized by the continuous creation of vast amounts of data, originated from different sources, and with different formats. First, with the rise of the *social networks* and, more recently, with the advent of the *Internet of Things* (IoT), in which everyone and (eventually) everything is linked to the Internet, data with enormous potential for organizations is being continuously generated. In order to be more competitive, organizations want to access and explore all the richness that is present in those data. Indeed, Big Data is only as valuable as the insights organizations gather from it to make better decisions, which is the main goal of Business Intelligence. In this paper we describe an experiment in which data obtained from a NoSQL data source (database technology explicitly developed to deal with the specificities of Big Data) is used to feed a Business Intelligence solution.

Keywords

Big Data, NoSQL, Business Intelligence, Dashboard, Pentaho.

1. Introduction

In the last decades we have witnessed an increase in the volume of data that is produced by organizations and by people in their daily life activities. In the latter case, as a result of the boom occurred with social networks, increasing amounts of data are being generated by people, either by themselves and/or as a result of the interaction with other people. These new data have great potential for organizations as a source of insight about people needs, opinions, market tendencies, and so on.

According to IBM estimations, as of 2012, every single day 2.5 Exabyte (2.5 X 10^{18} bytes) of data were generated. In the near future, with the so-called Internet of Things (IoT), in which virtually any electronic device with processing capacity will be integrated in the Internet, generating and consuming data, the amount of data we will have to deal with will increase dramatically.

These new data comes in larger amounts, at higher rates, from different sources, and with distinct features. In this context one might distinguish among three kinds of data to store and process (Halper & Krishnan, 2013):

Structured data – data with a rigid and previously known structure, in which all elements share the same format and size. This is the kind of data, traditionally found in business applications, that have been stored in relational databases;

Semi-structured data – data with a high degree of heterogeneity, which is not easily represented in fixed data structures. Typically, these kind of data have been stored using specific languages such as XML (*Extensible Markup Language*) data, RDF (*Resource Description Framework*) data, and so on;

Unstructured data – data without a structure, such as text, video, or multimedia content. In this group one can find the kind of data which have grown exponentially in the last decade, with some estimates pointing that, nowadays, 80% to 90% of the generated data is unstructured data. Examples include documents, images, photos, email messages, webpages, and so on.

In a few words, this is what characterizes the era of the Big Data: huge amounts of both structured and unstructured data, produced and consumed at increasing higher rates. These new features constitute an enormous challenge to the more traditional relational database technology. To answer to the new challenges created by Big Data, a new family of database technologies has emerged – the NoSQL databases.

In the present there are four families of NoSQL databases (Document, Column, Key/Value and Graph databases), each one of them with their own characteristics, strengths and weaknesses, but all sharing the same goal: to deal with the new challenges brought by Big Data (Cunha & Pereira, 2015) (Sousa & Pereira, 2015).

Despite their youth, NoSQL databases are becoming major players in the database market. For instance, the well-known DB-Engines Ranking (http://db-engines.com/en/ranking), which ranks databases according to their popularity, puts three NoSQL databases in the top 10: MongoDB, Cassandra and Redis.[1]

With Big Data organizations understood the enormous potential underlying those vast amounts of available data. They only had to use the right tools to treat those data, in order to better understand their business and their markets. Business Intelligence (BI) tools are what organizations need to access those data and extract the insights needed to make the best decisions and outshine their competition.

By definition, BI is the collection of methods and tools that allow organizations to transform data into valuable information to support decision-making (Kimball & Ross, 2013). Since data may come from different sources and in a multitude of formats, BI tools need to have the capacity to *Extract* data from those sources, to *Transform* those data (selecting, cleaning, joining, calculating, coding/decoding, etc.) according to the purpose of the solution, and to *Load* the data into a repository commonly known as Data Warehouse (DW). In addition to the ETL capabilities, BI tools provide the mechanisms to build suitable information delivery front-ends for decision makers, such as reports and dashboards (see Figure 1).

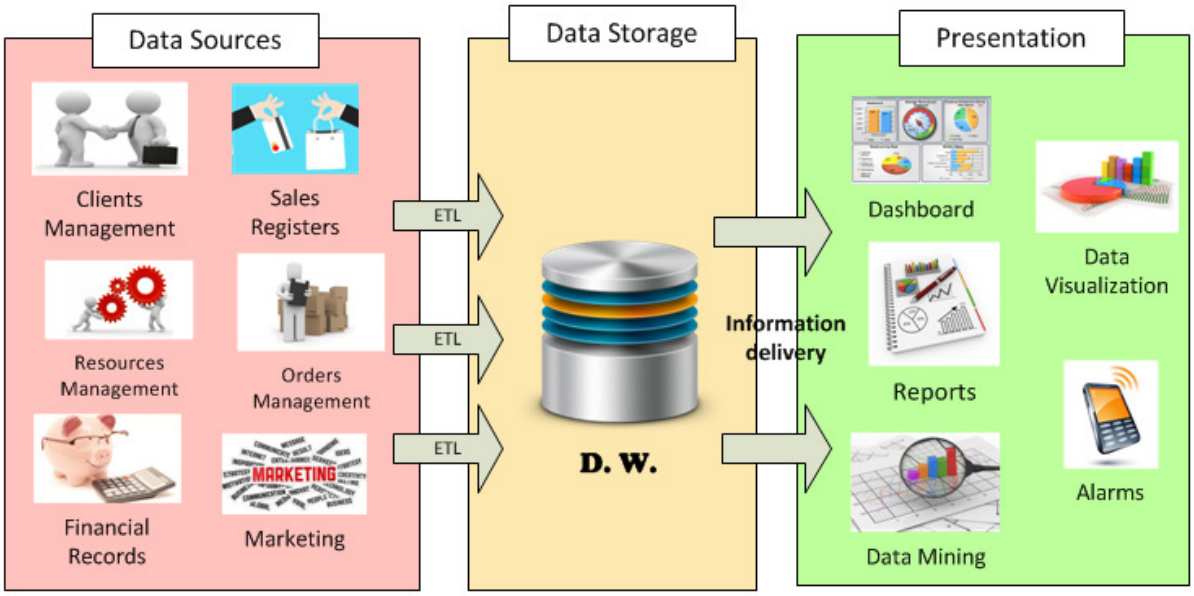


Figure 1. Global Perspective of a BI Solution

In this paper, we describe the development of a BI solution, which is being used by a Portuguese firm (Movvo). This firm deals with the detection and monitoring of client movements in shopping spaces, such as malls. The purpose of the BI solution is to provide decision makers with information about the habits of the shoppers, time spent in shopping, locals visited, etc., which is needed for them to decide how to organize the space.

[1] Site accessed in October, 2015.

Concerning the structure of this paper, after a very brief presentation of the main concepts around Big Data and the database technology that promises to solve its major challenges - NoSQL databases, we made a very concise introduction to the area of Business Intelligence, stressing its value to support decision-making in organizations. In the next sections, we describe a development project in which data captured from a NoSQL database is used to feed a specific BI solution. To begin with, we describe the real context in which the BI solution was conceived and then we quickly advance to its development. We divided this project in two parts: the first part deals with the extraction, transformation and loading (ETL) of the NoSQL data into a local database; the second part of the project involves the construction of a dashboard to present data in order to support decision-making. Finally, some conclusions about the project are presented and future work is envisaged.

2. Context of the Experiment

In order to better manage a shopping mall, decision makers would like to know simple facts such as "how many visitors walk by a shop?", "how many visitors enter a shop?", "how many visitors did made an acquisition?", "which are the busiest and the quietest hours?", "How much time shoppers spent in a shop?", "which are the locals most visited in a shop?", and so on. In order to accomplish that, a system for the detection and monitoring of people movements in space must be in place. Luckily, nowadays almost everyone use mobile phones so, making use of the GSM technology (*Global System for Mobile Communications*), in particular using the IMEI (*International Mobile Equipment Identity*) and the IMSI (*International Mobile Subscriber Identity*), one can easily trace the movements of people in a monitored space. This is a very convenient solution as those "mobile identifiers" are never switched off (mobile phones only stop emitting a signal if their battery is removed). Therefore, with a convenient distribution of sensors in a given space one can trace the movements of people in that area.

The data used in this project were obtained mostly through sensors installed in a sporting goods store located in a shopping mall. Data are captured by Movvo and stored in a NoSQL database (in this case, a Cassandra system), all day long, every day of the week, non-stop. Using an API (*Application Program Interface*), the Cassandra database provides access to the data in the JSON format (*JavaScript Object Notation*), which is a very simple a convenient format. These data are used to feed the developed BI solution (see Figure 2).

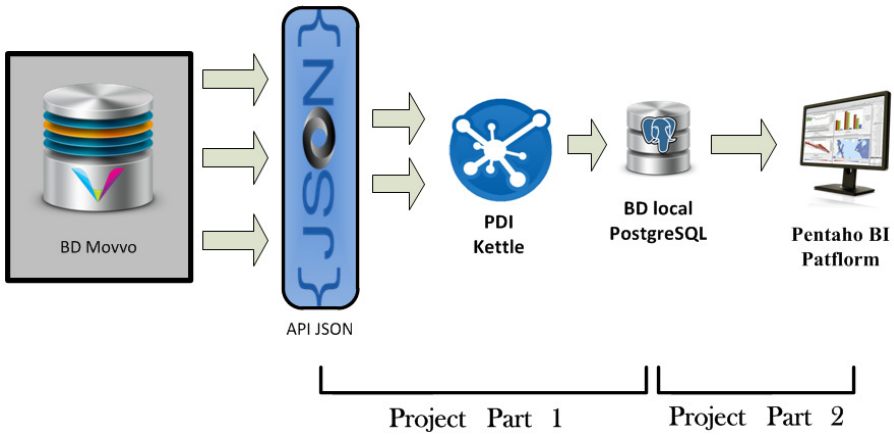


Figure 2. Architecture of the BI Solution

The development of the BI solution involves two parts. In the first part, data are retrieved from Cassandra using the provided API and, after some processing tasks, are stored in a local database. In the second part a suitable dashboard is developed, according to the needs of decision makers.

Regarding the technologies used in the development of the BI solution, in addition to the PostgreSQL used to manage the data repository, we selected the Pentaho family of products. In particular:

- Pentaho Data Integration (Kettle) – the solution offered by Pentaho for ETL. In this project Kettle was used to extract data from the provided Cassandra API, do the necessary treatments, and store the resulting data in the PostgreSQL database (PDI, 2015);

- Pentaho BI Platform – A platform that allows us to take the data from the repository and turn it into useful information for decision makers, by providing tools for creating reports and information panels, or dashboards (PBIP, 2015).

In the next section, the first part of the project is described.

3. The Business Intelligence Solution – Part I

In order to develop a BI solution adequate to the needs of decision makers, regarding the management of a sporting goods store located in a shopping mall, we need to access data from which we may extract some metrics. The following table (Table 1) summarizes those data.

Table 1. Data needed for Decision-Making

Metric	Description
space-tickets	Number of registered sales in store
space-walk-bys	Number of detected people passing in front of the store
space-visitors	Number of detected people inside the store
space-visiting-time	Average duration of visits to the store
zone-visitors	Number of detected people in each zone of the store
weather	About the weather and temperature

To get those data using the provided API, a request such as the following has to be made:

```
https://(...)/days/2015-08-15T00:00:00Z/2015-08-16T23:00:00Z?metrics=space-
visitors:hour:series
```

In this example we issued a request to search for data about the number of visits to that store from 15 of August to 16 of August of 2015. The resulting response is as follows (JSON format):


```
{
  ▾ "space-visitors:hour:series": [
      ▸ {… },
      ▾ {
            date: "2015-08-15T12:00:00+01:00",
            value: 346
        },
      ▾ {
            date: "2015-08-15T13:00:00+01:00",
            value: 322
        },
      ▾ {
            date: "2015-08-15T14:00:00+01:00",
            value: 428
        },
      ▸ {… },
```


As we can see, 346 visits were detected in the store at Noon on 15 of August of 2015. In fact this represents a total of visits to the store between 11.00am and Noon; 322 visits between the Noon and the 1.00pm; and 428 visits between 1.00pm and 2.00pm. The result extends through the remaining hours of the day, in the two days requested.

The API provided allows us to group various metrics in a single request, simply by writing the request as follows:

```
https://(...)/days/2015-08-15T00:00:00Z/2015-08-25T23:00:00Z?metrics=space-
tickets:hour:series,space-walk-bys:hour:series,space-visiting-
time:hour:series,space-visitors:hour:series,zone-visitors:day:series
```


To obtain data from Cassandra, and to process and store them into a local database (PostgreSQL) a set of ETL steps were developed in Kettle. This specific ETL was developed in order to be autonomous, that is, it does not require the user to enter the dates in the requests to the API provided. In Figure 3 we can see the developed ETL steps, used to extract data from the provided API, do the necessary transformations, and finally load/refresh the data in the local database (tables Zones, Dates, Hour_records and Meteo).

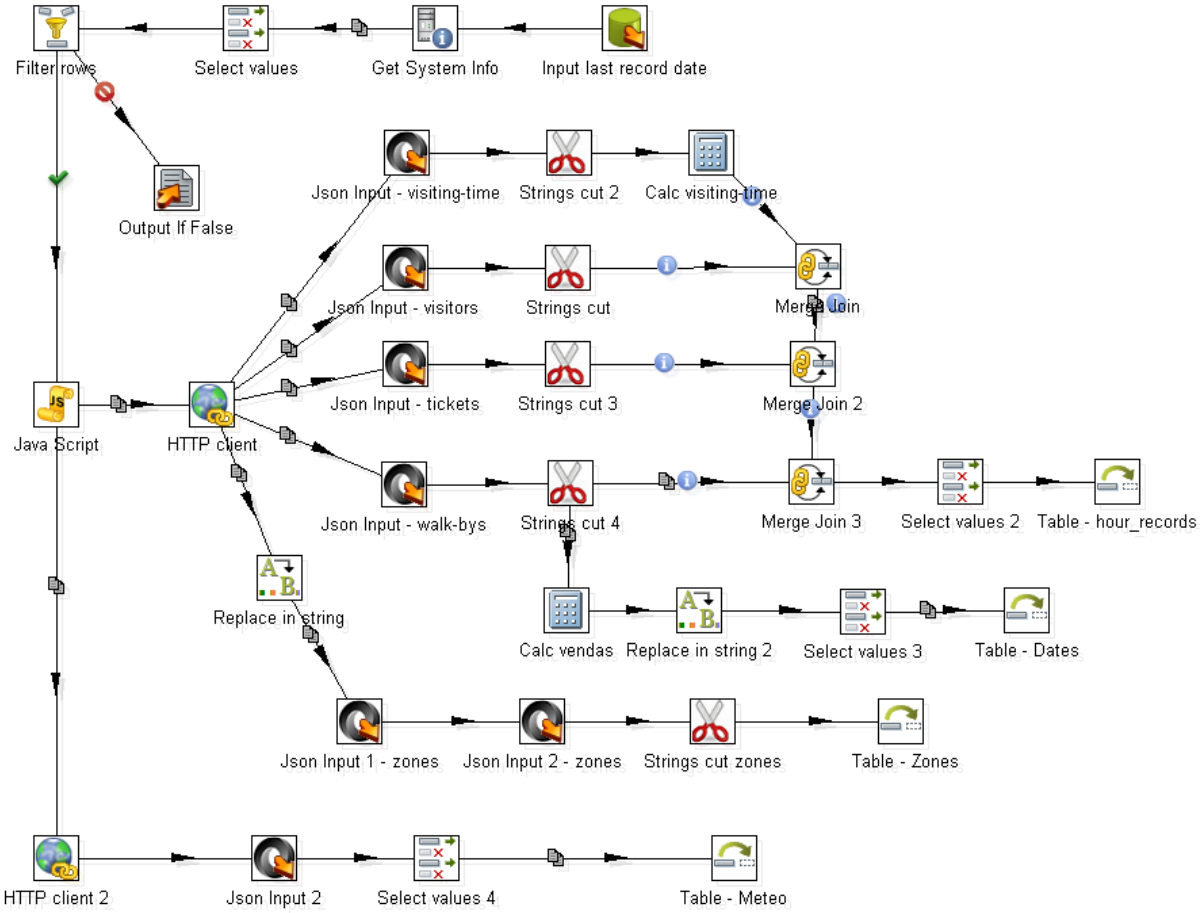


Figure 3. The ETL used to load and refresh the local PostgreSQL database tables

Unfortunately, due to space limitations, this is not the place to explain each one of the steps in the ETL above. Anyway, the experience using the graphical interface of Kettle (named Spoon) to develop ETL has been quite interesting and rewarding. Spoon has a wide range of steps, such as Data Input and Output, Statistics, Validation, Mapping, Utilities, and so on, which may be added to the workspace in a drag-and-drop fashion.

4. The Business Intelligence Solution – Part II

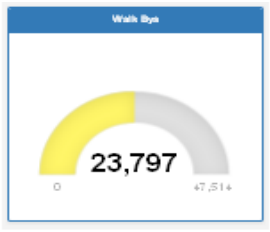
The second part of the development of the BI solution involves the exploitation of the Pentaho BI Platform tool to visually display the data previously collected through a dashboard. This dashboard is composed of several components, such as graphics, tables and even a map of the store. It was developed using a tool named CDE (*Community Dashboard Editor*), an open source tool designed to simplify the creation and editing of dashboards.

The first thing to do in the development of a dashboard is to identify “what” information we would like to see and “how” it should be displayed. In this case, the dashboard must include the following elements, among others:

1. The value of the metrics 'space-walk-bys', 'space-visitors' and 'space-tickets', compared to the maximum recorded in a given month;
2. The relation between the metrics 'space-visitors' and 'space-tickets' in the form of an area chart;
3. A 'Pie Chart' graphic and table with the data about the zones and correspondent 'zone-visitors';
4. Overview of the metrics 'space-walk-bys', 'space-visitors' and 'space-tickets' over a given year;
5. List of the 10 weeks in which there were more sales and its comparison with the number of visits.

Once again, due to space limitations, we only present some of the components of the dashboard. For instance, regarding the first of the elements in the dashboard, one can use a 'Gauge Component' to visualize each of the metrics required. With this component the metric value is displayed within a range, thus giving the user a better sense of the magnitude of its value. The range is set between 0 and the maximum value recorded in the current month. This component was built to be feed with eight values: the title, the value of the metric, the minimum value of the scale, the maximum value of the scale, the main color of the component, the color for the minimum value, the color for the medium value, and the color for the maximum value. In Figure 4 we show the aspect of this component used to visualize the metric 'space-walk-bys', and the SQL code used in the data source.

a)



b)

```
Select
    CONCAT(' ') as title,
    aux2.walk-bys,
    CONCAT('0') as min,
    MAX(aux.walk-bys),
    CONCAT('#E0E0E0'),
    CONCAT('#FFF567'),
    CONCAT('#FFF567'),
    CONCAT('#FFF567')
from (
    select hour_records.date, sum(hour_records.space-walk-bys) as walk-bys
    from hour_records,dates
    where dates.date=hour_records.date and month in
                (select month from dates where date=${pdate})
        group by hour_records.date
        ) as aux,
        (select sum(outdoor) as walk-bys
    from hour_records
    where date=${pdate}
    group by hour_records.date) as aux2
group by aux2.walk-bys
```

Figure 4. a) The component of the dashboard; b) The SQL to feed it

An example of a component to illustrate the second element of the dashboard appears in Figure 5. an area chart to relate the metrics' space-tickets' and 'space-visitors' along with three text components, highlighting the hours and the maximum and minimum values recorded for the metric' Space-visitors' and the average value of the selected day.

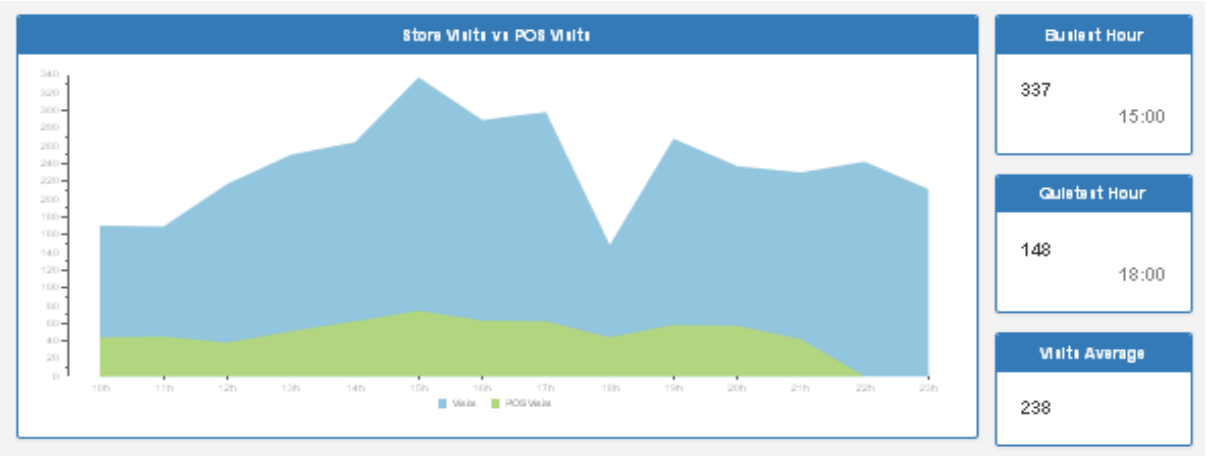


Figure 5. The relation between the metrics 'space-visitors' and 'space-tickets'

The third element of the dashboard was materialized using the component in Figure 6, below. Where we can see a Pie Chart indicating the zones of the store and the number of visitors in those zones during a given day. The table on the right list the zones in descending order of the number of visitors.

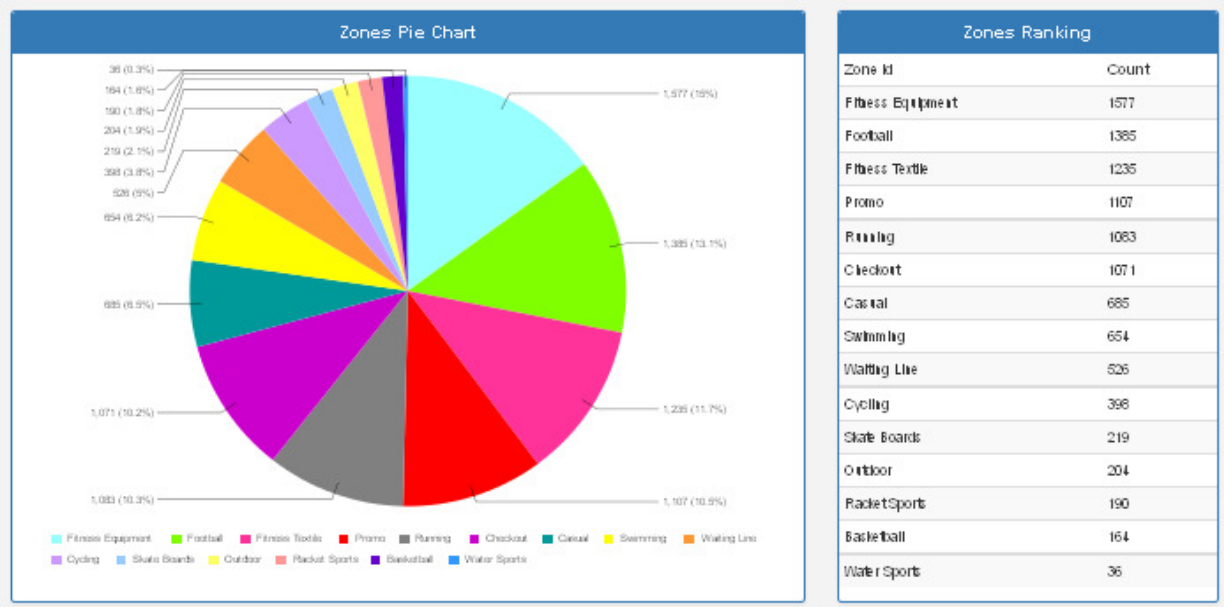


Figure 6. The zones of the store and their visitors in a given day

The fourth element of the dashboard allows the users of the BI solution a more comprehensive analysis of the trend of the metrics 'space-walk-bys', 'space-visitors' and 'space-tickets' throughout the year, the dashboard included a line chart with the data of a year (Figure 7). For a more detailed view this component allows us to display only one of the metrics individually, by simply hiding the others.

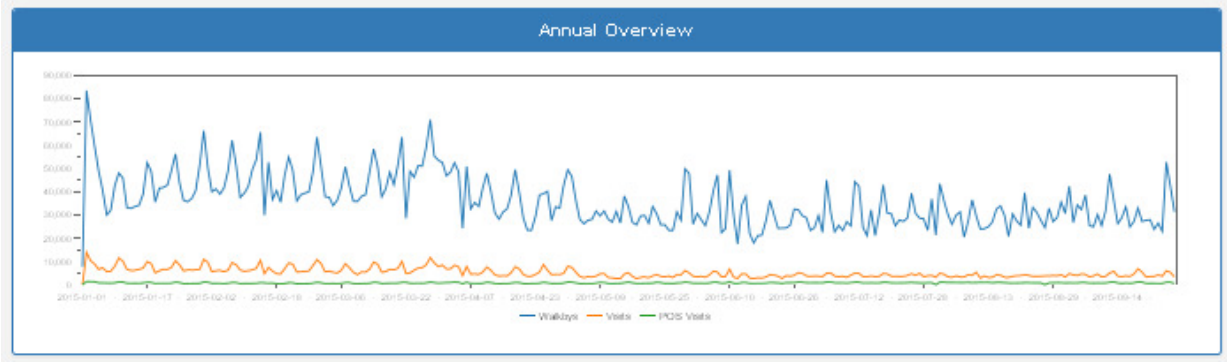


Figure 7. Line Graph with the evolution over a year of the metrics 'space-walk-bys', 'space-visitors' and 'space-tickets'

Finally, the fifth element of the dashboard proposes to illustrate the 10 weeks in which there were more sales, making a comparison with the number of visitors in those periods.

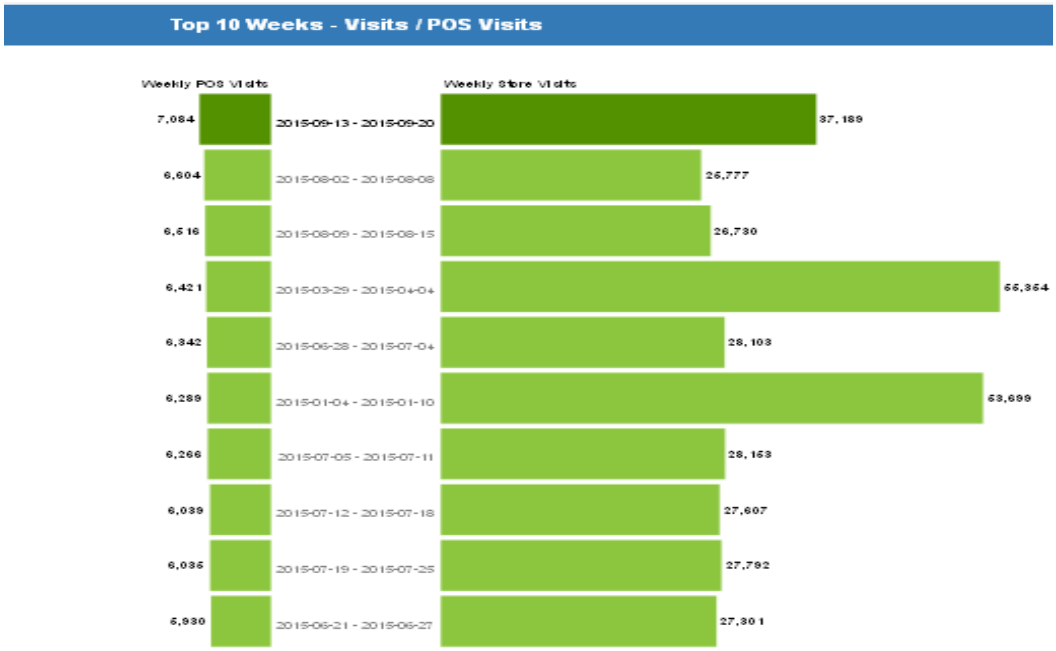


Figure 8. The Top Ten sales weeks and their correspondent number of visitors

This ends the second part of the BI solution development project, with a dashboard which includes five components that allow decision-makers to better understand what happens in the store and so, making them more able to manage their business.

5. Conclusions

In this paper we present an experiment regarding the development of a BI solution which gets its data from a NoSQL source. The context of the project was a sporting goods store located in a shopping mall, in which the movement of visitors, around and inside the store, as long as the time spent in each zone of the stores was monitored and registered in a Cassandra system. Those data, completed with data about sales, allowed the definition of several metrics in order to understand the behavior of visitors. Thus, a BI solution was developed.

We claim that dashboards are a great way to give responsible managers a systematic way to easily access the most relevant information in order to gain insight and to better support decision-making. The tools used have proved to be suitable for implementing the project. Unfortunately, it has not allowed to take full advantage of the ETL tool, since the data used had been previously processed and was not in raw state.

In the case of BI Platform tool, despite the wide range of available components to the community not all of the desired components were available. This has not proved to be an obstacle, as in the case of an open source environment it was possible, using an existing component, to edit it to create other functionalities.

References

Cunha, J.P., & Pereira, J.L. (2015). Column-Based Databases: Estudo Exploratório no Âmbito das Bases de Dados NoSQL. Proceedings da 15ª Conferência da Associação Portuguesa de Sistemas de Informação. CAPSI 2015. Lisboa.

Halper, F., & Krishnan, K. (2014). TDWI Big Data Maturity Model Guide Interpreting Your Assessment Score. TDWI Benchmark Guide 2013–2014.

Kimball, R., & Ross, M. (2013). The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition, Wiley.

PBIP. (2015). "Pentaho Business Intelligence Platform" (http://www.pentaho.com/product/business-visualization-analytics; accessed in June of 2015).

PDI. (2015). "Pentaho Data Integration - Kettle ETL tool" (http://etl-tools.info/en/pentaho/kettle-etl.htm; accessed in April of 2015).

Sousa, G., & Pereira, J.L. (2015). Document-Based Databases: Estudo Exploratório no Âmbito das Bases de Dados NoSQL. Proceedings da 15ª Conferência da Associação Portuguesa de Sistemas de Informação. CAPSI 2015. Lisboa.

