Pentaho and Jaspersoft: A Comparative Study of Business Intelligence Open Source Tools Processing Big Data to Evaluate Performances

Victor M. Parra
School of Computing and Mathematics
Charles Sturt University
Melbourne, Australia

Azeem Mohammad
School of Computing and Mathematics
Charles Sturt University
Melbourne, Australia

Ali Syed
School of Computing and Mathematics
Charles Sturt University
Melbourne, Australia

Malka N. Halgamuge
School of Computing and Mathematics
Charles Sturt University
Melbourne, Australia

***Abstract*—Regardless of the recent growth in the use of "Big Data" and "Business Intelligence" (BI) tools, little research has been undertaken about the implications involved. Analytical tools affect the development and sustainability of a company, as evaluating clientele needs to advance in the competitive market is critical. With the advancement of the population, processing large amounts of data has become too cumbersome for companies. At some stage in a company's lifecycle, all companies need to create new and better data processing systems that improve their decision-making processes. Companies use BI Results to collect data that is drawn from interpretations grouped from cues in the data set BI information system that helps organisations with activities that give them the advantage in a competitive market. However, many organizations establish such systems, without conducting a preliminary analysis of the needs and wants of a company, or without determining the benefits and targets that they aim to achieve with the implementation. They rarely measure the large costs associated with the implementation blowout of such applications, which results in these impulsive solutions that are unfinished or too complex and unfeasible, in other words unsustainable even if implemented. BI open source tools are specific tools that solve this issue for organizations in need, with data storage and management. This paper compares two of the best positioned BI open source tools in the market: Pentaho and Jaspersoft, processing big data through six different sized databases, especially focussing on their Extract Transform and Load (ETL) and Reporting processes by measuring their performances using Computer Algebra Systems (CAS). The ETL experimental analysis results clearly show that Jaspersoft BI has an increment of CPU time in the process of data over Pentaho BI, which is represented by an average of 42.28% in performance metrics over the six databases. Meanwhile, Pentaho BI had a marked increment of the CPU time in the process of data over Jaspersoft evidenced by the reporting analysis outcomes with an average of 43.12% over six databases that prove the point of this study. This study is a guiding reference for many researchers and those IT professionals who support the conveniences of Big Data processing, and the implementation of BI open source tool based on their needs.**

Keywords*—Big Data; BI; Business Intelligence; CAS; Computer Algebra System; ETL; Data Mining; OLAP

I. Introduction

Business Intelligence software converts stored data of a company's clientele profile and turns it into information that forms the pool of knowledge to create a competitive value and advantage in the market it is in [1]. Additionally, Business Intelligence is used to back up and improve the business with reasonable data and use the analysis of this data, to continuously improve an organisation's competitiveness. Part of this analysis is to provide timely reports, for management's to make the decision based on factual information, so their decision-making is based on concrete evidence. Howard Dresner, from Gartner Group [2], was the first to coin the term Business Intelligence (BI), as a term to define a collection of notions and procedures to support the decision-making, by using information found upon facts.

BI system gives enough data to use and evaluate the needs and desires of customers and in addition it allows to: [3]: i) Design reports for departments or global areas in a company, ii) Build a database for customers, iii) Create scenarios for decision-making, iv) Share information between areas or departments of a company, v) Sandbox studies of multidimensional designs, vi) Extract, transform and process data, vii) Give a new approach to decision-making and viii) Improve the quality of customer service.

The benefits of systemizing BI include the amalgamation of information from several sources. [4], creating user profiles for information management, reducing the dependence on the systems department, the reduction in the time of obtaining information, improves the analysis, and also improves the availability to access real-time information according to specific current business criteria.

The recent publication of Gartner Magic Quadrant for Business Intelligence Platforms 2015 [5] has highlighted the changes being taken by the BI sector to rapidly deploy platforms that can be used by both business users and analysts to extract information from collected data. Traditionally, business intelligence has been understood as a set of methodologies, applications and technologies used to transform data into information and then information into a personal profile of clients that is generated into structured data to serve different areas of business enterprise [6].

Therefore, Big Data will aid to develop better procedures that allow (BI) tools to be used to gather information, such as [7]: i) Process and analyse volumes of information; ii) Increase the universe of data to consider when decision-making: and inherent historical data of the company, to incorporate data from external sources ; iii) Provide an immediate response to the continued provision of real-time data of the devices and the possibilities of interconnections between devices; iv) Working with structures of complex and heterogeneous data: logs, emails, conversations, locations, voice, etc.; v) and lastly, to Isolate from the physical constraints of storage and process by making use of scalable solutions and high availability at a competitive prices.

This paper presents an experimental analysis of the comparison of two of the best positioned open source BI systems in the market: Pentaho and Jaspersoft, processing Big data and focussing on their Extract Transform and Load (ETL) and reporting processes by measuring their performance using Computer Algebra Systems. The aim of this paper is to analyse and evaluate these tools and outline how they improve the quality of data, and inadvertently helps us understand the market conditions to make future predictions base on trends.

Section II describes the capabilities and components of both Pentaho and Jaspersoft BI Open Sources. Section III introduces the computer algebra systems SageMath and Matlab. This is followed by the materials and methods (Section IV) used in the analysis and experimentation, especially the ETL and Reporting measurements and how they were implemented. In Section V, the results of the study for *CPUtime* as a function of the "size" from the input data for the ETL and Reporting processes from both Pentaho and Jaspersoft Business Intelligence Open Sources, applying two different Computer Algebra Systems. Section VI contains the discussion of the experimentation. Section VII, the conclusion of the study.

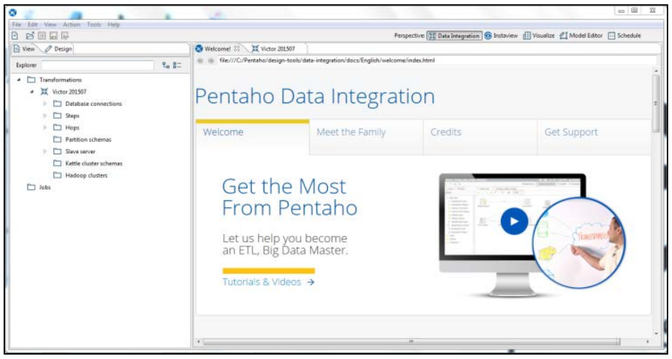
II. Pentaho and Jaspersoft Business Intelligence Open Sources

A. Pentaho

Pentaho, created in 2004 is the current leader of Business Solutions Intelligence Open Source. It offers its own solutions across the spectrum of resources to develop and maintain the operations of BI projects from the ETL with data integration to the dashboards with Dashboard Designer [8]. Pentaho has built its solution Business Intelligence integrating different existing and recognized solvency projects. Data Integration was previously known as Kettle; indeed, it retains its old name as a colloquial name. Mondrian is another component of Pentaho that retaining its own entity.

Pentaho has the following components:

a) ETL: Pentaho Data Integration (previously Kettle) is one of the most widely used ETL solutions and better valued in the market [9]. It has a long history, solidity, and robustness that make it a highly recommended tool. It allows transformations and works in a very simple and intuitive way, as it is shown in Fig. 1. Likewise, the Data Integration projects are very easy to maintain.

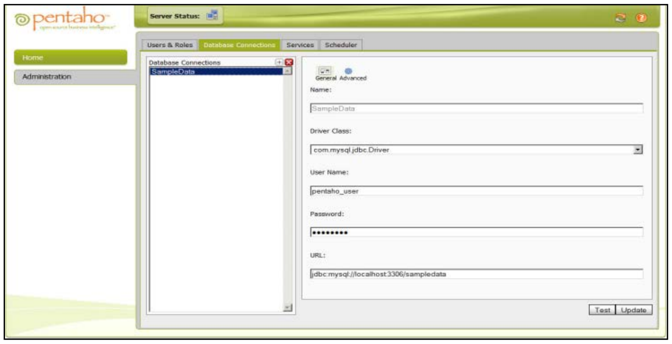


Get the Most From Pentaho

Let us help you become an ETL, Big Data Master.

Fig. 1. Pentaho Data Integration Interface, the ETL solution allows transformations and works in a very simple and intuitive way

b) Web Application-BI Server: The BI Pentaho Server is a 100% Java2EE allows us to manage all BI resources [10]. It has a BI user interface available where reports are stored, OLAP views and dashboards as it is illustrated in Fig. 2. In addition, it offers access to a management support that allows managing and monitoring both application and usage.



SampleData

Name: SampleData

Driver Class: com.mysql.jdbc.Driver

User Name: pentaho_user

Password: ••••••••

URL: jdbc:mysql://localhost:3306/sampledata

Fig. 2. Pentaho Server User Interface to manage BI resources where all the reports are founded, OLAP *views, and dashboards.* Also the access to a management supports that allows managing and monitoring both application and usage

c) Pentaho Reporting: Pentaho provides a comprehensive reporting solution. Covering all aspects needed in any reporting environment, as shown in Fig. 3. The Pentaho reporting tool is the old form of JFreeReport [11]: i) It provides a tool for reporting (Pentaho Reporting), ii) Provides an execution engine, iii) Provides Metadata tool for conducting reports Ad-hoc, and iv) Provides a user interface that allows ad-hoc reports (WAQR).

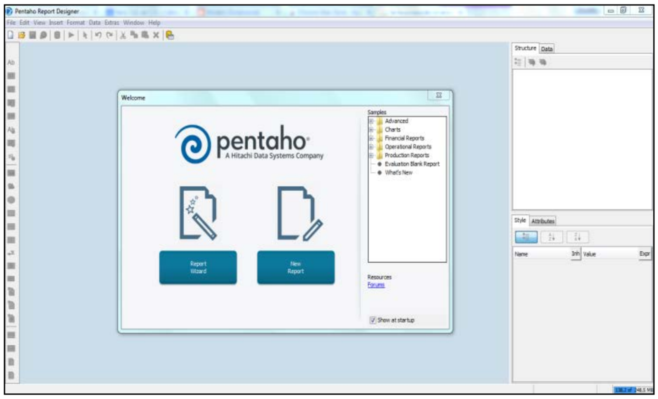


Fig. 3. Pentaho Reporting Interface with a comprehensive reporting solution, covering all aspects needed in any reporting environment

d) OLAP Mondrian: Online Analytical Processing is the technology that allows us to organize information in a dimensional structure that will allow us to move information by scrolling through its dimensions [12]. Mondrian is the Pentaho OLAP engine. Although it can be integrated independently on any other platform, and indeed it is the component. Data Integration that is used independently. Mondrian is a Hybrid OLAP engine that combines the flexibility of ROLAP engines with a cache that provides speed.

- Viewer OLAP: Pentaho Analyser: OLAP Viewer that comes with the Enterprise version [13]. Modern and easier to use than JPivot as it is illustrated in Fig. 4. AJAX provides an interface that allows great flexibility when creating the OLAP views.

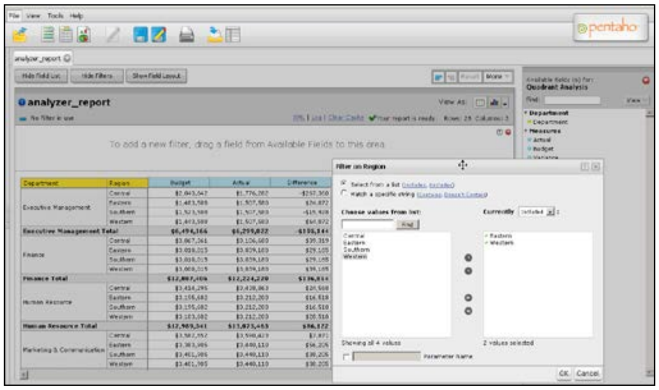


Fig. 4. Pentaho Analyser Interface to create OLAP views that *p*rovide a comprehensive reporting solution. Covering all aspects needed in any reporting environment

e) Dashboards: Pentaho provides the possibility of making dashboards [13] through the web interface using the dashboard designer as it is shown in Fig. 5.

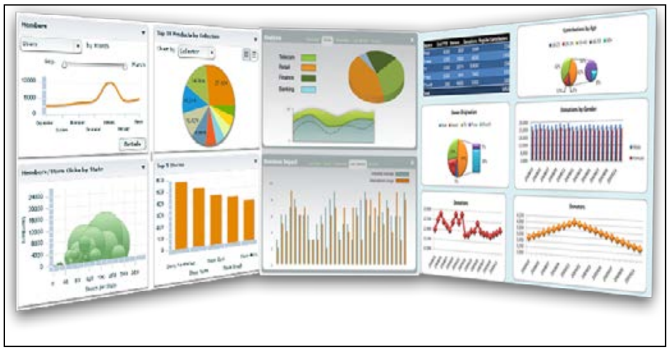


Fig. 5. Pentaho Dashboards Interface that provides the possibility of making dashboards through the web interface by using the dashboard designer

B. Jaspersoft

Jaspersoft is the company behind the famous and extended Jaster Reports. Open Source reporting solution preferred by most developers to embed in any Java application that requires a reporting system. Jaspersoft has built its solution B.I. around its reporting engine [14]. This has been done differently from Pentaho. Jasper has integrated its projects that also solves existing and consolidates projects nonetheless, has not absorbed it. This strategy makes it "depend" on Talend solution regarding ETL and Mondrian - Pentaho for the OLAP engine. Jasper has access to the code Mondrian that can adapt and continue its developments with Mondrian.

Jaspersoft has the following components:

a) ETL - JasperETL is actually Talend Studio. Talend, unlike Kettle, it has not been absorbed by Jasper and remains an independent company that offers its products independently [15]. Working with Talend is also quite user-interface intuitive and proprietary although the approach is completely different. Talend is a code generator that is the result of an ETL exercise and it is native Java or Perl code. It can also compile and generate Java procedures or instructions. Talend is more oriented to a type of programmer used with a higher level of technical expertise than it requires by Kettle as it is illustrated in Fig. 6. To sum up, the flexibility is much better with this approach.

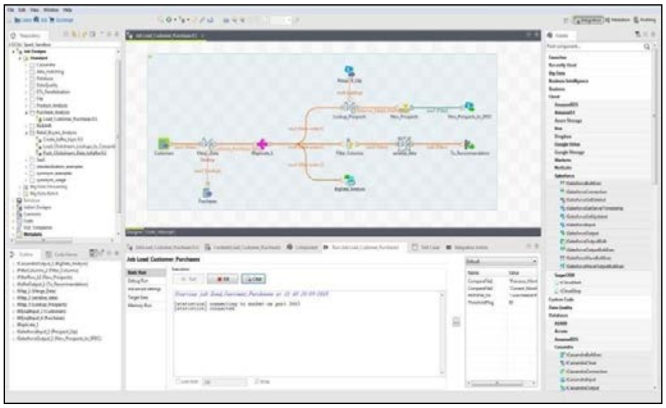
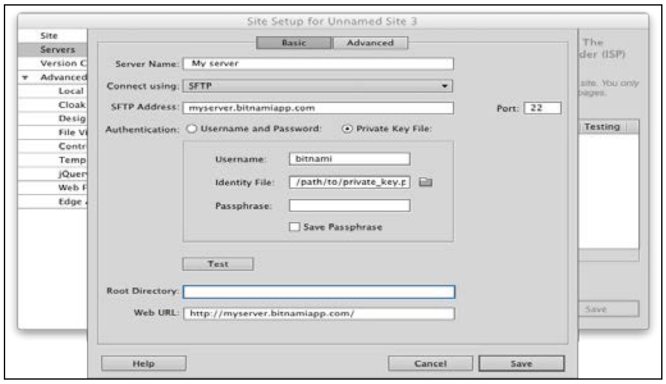


Fig. 6. Jaspersoft ETL Interface is actually Talend Studio, it is also quite user-interface intuitive and a code generator

b) Web Application–JasperServer: JasperServer is a 100% Java2EE that allows us to manage all BI resources [16]. The overall look of the web application is a bit minimalist without sacrificing the power as shown in Fig. 7. However, having all resources available on the top button bar makes it a 100% functional application and has all the necessary resources for BI.



Server Name: My server

Connect using: SFTP

SFTP Address: myserver.bitnamiapp.com

Port: 22

Authentication: ○ Username and Password: ● Private Key File:

Username: bitnami

Identity File: /path/to/private_key.p

Passphrase:

□ Save Passphrase

Root Directory:

Web URL: http://myserver.bitnamiapp.com/

Fig. 7. Jaspersoft Server Interface *is a 100%* Java2EE to manage business intelligence resources

c) Reports: As described, the report engine is the solution of Jaspersoft as it is illustrated in Fig. 8. The component provides features such as i) Report development environment: Ireport as a system based on environment NetBeans. What makes it challenging to machine resources? In return it offers great flexibility, ii) System of metadata (Domains) the web. These, along with ad-hoc reports, are the strengths of this solution, iii) Web Interface for ad-hoc reports really well resolved, iv) The runtime JasperReports widely was known and used in many projects where a solvent reporting engine is needed, and v) The reports can be exported into PDF, HTML, XML, CSV, RTF, XLS and TXT.

- *Predefined – Ireport:* IReport is a working environment that allows a large number of features [17]. Here something like that Talend is a working environment with larger demands as a result of offering a number of possibilities occurs.
- *Ad hoc*: This is the real strength of Jasper solutions. The editor of ad-hoc reports is the best structured and best featured tool for analysing [17]. It offers: i) Selection of different types of templates and formats, ii) Selection of different data sources, iii) Validation consultation on the fly, iv) Creation of reports by dragging fields to the desired location: i) Tables, ii) Graphics, iii) Crosstable (Pivot), and iv) Edition of all aspects of the reports.

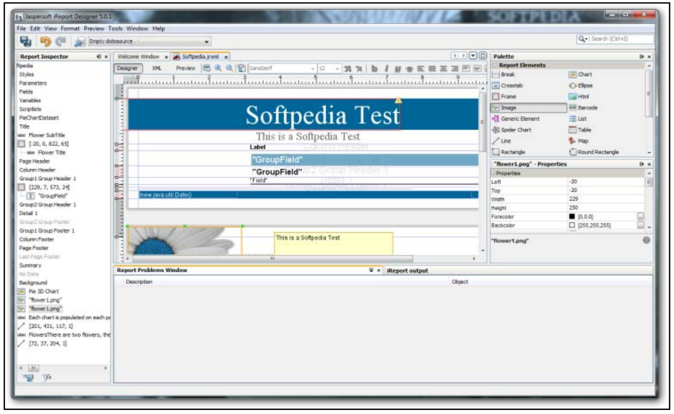
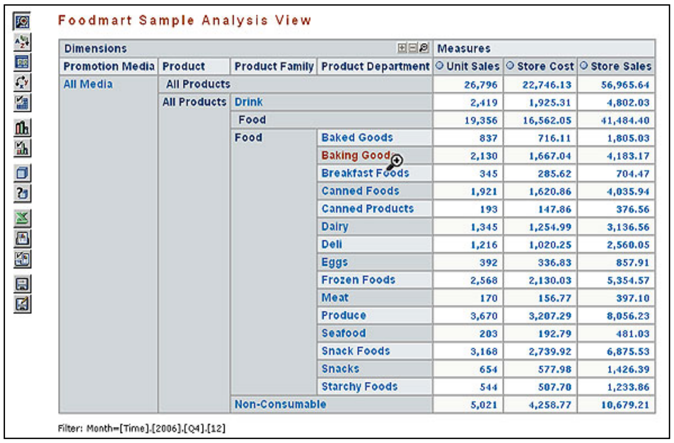


Fig. 8. Jaspersoft Reporting Interface that includes report development environment, a system of metadata (Domains) web, web interface for ad-hoc reports and runtime JasperReports

d) OLAP: The OLAP engine that uses JasperServer is Mondrian and uses a Viewfinder-JasperAnalysis [18], which is no longer JPivot but with a layer of makeup as shown in Fig. 9. Already mentioned in Pentaho paragraph.



Foodmart Sample Analysis View

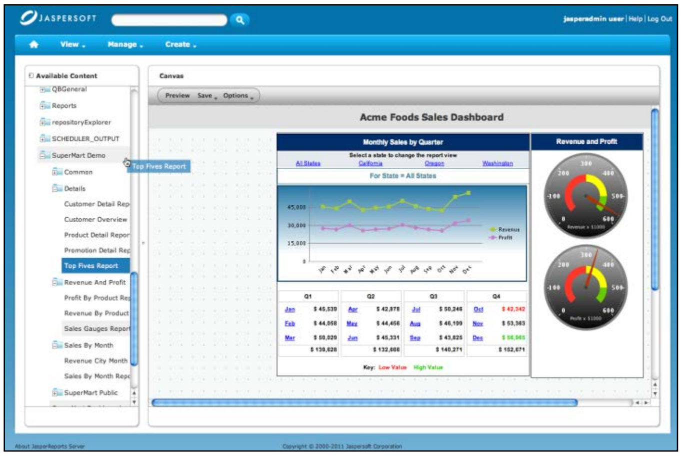
Dimensions				Measures		
Promotion Media	Product	Product Family	Product Department	○ Unit Sales	○ Store Cost	○ Store Sales
All Media	All Products			26,796	22,746.13	56,965.64
	All Products	Drink		2,419	1,925.31	4,802.03
		Food		19,356	16,562.05	41,484.40
		Food	Baked Goods	837	716.11	1,805.03
			Baking Goods	2,130	1,667.04	4,183.17
			Breakfast Foods	345	285.62	704.47
			Canned Foods	1,921	1,620.86	4,035.94
			Canned Products	193	147.86	376.56
			Dairy	1,345	1,254.99	3,136.56
			Deli	1,216	1,020.25	2,560.05
			Eggs	392	336.83	857.91
			Frozen Foods	2,568	2,130.03	5,354.57
			Meat	170	156.77	397.10
			Produce	3,670	3,207.29	8,056.23
			Seafood	203	192.79	481.03
			Snack Foods	3,168	2,739.92	6,875.53
			Snacks	654	577.98	1,426.39
			Starchy Foods	544	507.70	1,233.86
		Non-Consumable		5,021	4,258.77	10,679.21

Filter: Month=[Time].[2006].[Q4].[12]

Fig. 9. The OLAP engine that uses JasperServer is Mondrian and uses a Viewfinder-JasperAnalysis which is no longer JPivot but with a layer of makeup

e) Dashboards: Dashboard Designer. Illustrated in Fig.10.

- *Predefined:* They do not make much sense, given the designer panels [19]. In any case, to be a Java platform it can always include proper developments.
- *Ad-hoc:* Dashboard Designer: It is back to a really easy and simple use of the web editor.



Acme Foods Sales Dashboard

Monthly Sales by Quarter

Select a state to change the report view

All States California Oregon Washington

For State = All States

Q1		Q2		Q3		Q4	
Jan	$ 45,539	Apr	$ 42,878	Jul	$ 50,246	Oct	$ 42,342
Feb	$ 44,058	May	$ 44,456	Aug	$ 46,199	Nov	$ 53,363
Mar	$ 50,029	Jun	$ 45,331	Sep	$ 43,825	Dec	$ 58,965
	$ 139,628		$ 132,666		$ 140,271		$ 152,671

Key: Low Value High Value

Fig. 10. Jaspersoft Dashboards Designers Interface able to select predefined or ad-hoc

III. Computer Algebra Systems

A. Sagemath

SageMath is a computer algebra system (CAS) that is built on mathematical packages and contrasted as NumPy, Sympy, PARI / GP or Maxima. It accesses the combined power of the same through a common language based on Python. The interaction code combines cells with graphics, texts or formulas enriched with LaTeX rendered. Additionally, SageMath is divided into a core that performs calculations and an interface that displays and interacts with the user. Even, a command line-based text is also available using Python that allows interactive control calculations [20]. The Python programming language supports object-oriented expressions and functional programming. Internally, SageMath is written in Python and a modified version of Pyrex called Cython. It allows parallel processing [21] using both multi-core processors and symmetric multiprocessors. It also provides interfaces to other non-free software as Mathematica, Magma, and Maple (undistributed with SageMath) that allows users to combine software and compare results and performances.

All the packages cover most features such as i) Libraries of elementary and special functions, ii) 2D and 3D graphs of both functions and data, iii) Data manipulation tools and duties, iv) A toolkit for adding user interfaces to calculations and apply, v) Tools for image processing using Python and Pylab, vi) Tools to visualize and analyze graphs, vii) Filters for importing and exporting data, images, video, sound, CAD, and GIS, viii) Sage embedded in documents LaTeX6 [22].

B. Matlab

Matlab is a computer algebra system (CAS) that provides an integrated environment that develops and offers representative characteristics such as the implementation of algorithms, data representation and functions. Also, communication with programs in other languages and other hardware devices [23], among others are advanced. The Matlab package has two extra tools that extend those functionalities: Simulink is a platform for multi-domain simulation and GUIDE that is a graphic user interface - GUI. Additionally, its potential could be expanded using Matlab toolboxes; and Simulink blocks with block sets.

The language of Matlab is interpreted, and can run in both interactive environments through a script file (*.m files). This language allows vector and matrix operations to function, lambda calculus, and object-oriented programming. An additional tool called Matlab Builder has been launched that contains an "Application Deployment" which allows using Matlab functions, as library files, that provides the ability to be used with environments such as .NET or Java. Matlab Component Runtime (MCR) should be used on the same machine where the main application is set for the Matlab function properly [24]. One of the versatilities of this CAS is that it is quite useful to carry out measurements and it provides an interface to interact with other programming languages. Thus, Matlab can call functions or subroutines written in C or FORTRAN [25]. As the process is accomplished by, creating a wrapper function that allows them to be passed and returned by their data types Matlab.

IV. Materials and Methods

Accurately measuring the processing times is not a trivial task, and the results may vary significantly from one computer to another. The number of factors that influence the execution times has used an algorithm, operating system, processor speed, the number of processors and instruction sets that understands the amount of RAM, and cache, and speed of each, math coprocessor, GPU Among each other. Even on the same machine, the same algorithm sometimes takes much longer to give results, due to factors such as using more time than other applications, or if there is enough RAM when running the program.

The objective is to compare only the ETL and Reporting processes, trying to draw independent conclusions from one machine to another. The same algorithm can be called with different input data.

The goal of this study is to measure the run-time as a function of the "size" of the input data. For this, two techniques are used: - Measure run time of programs with different input data sizes and - Count the number of operations performed by the program.

A. ELT Measurement

With Sage was measured the run time and efficiency of ETL processes in both BI tools mentioned previously as it is illustrated in Fig. 11. For the CPU time, Sage uses the concepts of CPU time and Wall time [12], which are the times that the computer is dedicated solely to the program.

The following flow chart shows the ETL process measurement.

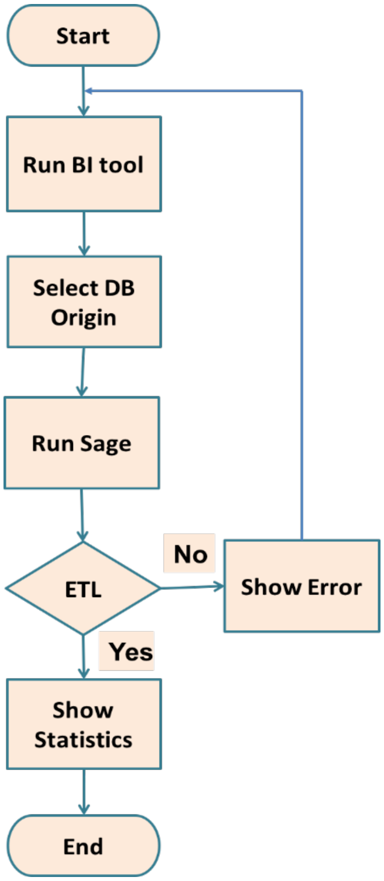
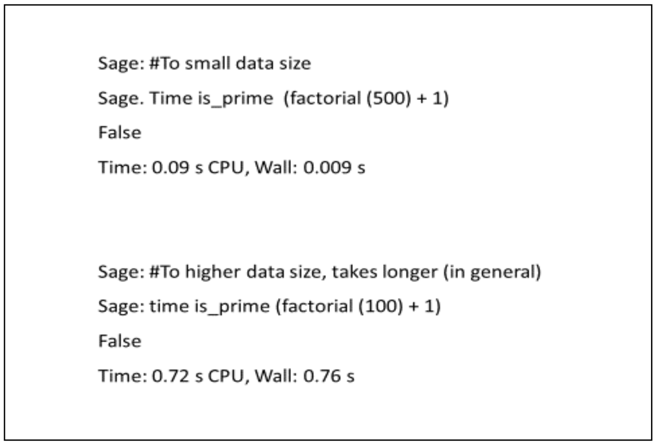


Fig. 11. The ETL process measurement using the Sage computer algebra system for all databases

The CPU time is dedicated to our calculations, and Wall time clock time between the beginning and the end of the calculations. Both measurements are susceptible to unpredictable variations. The easiest way to get the run time of a command is to put the word time to the command as it is shown in Fig. 12.

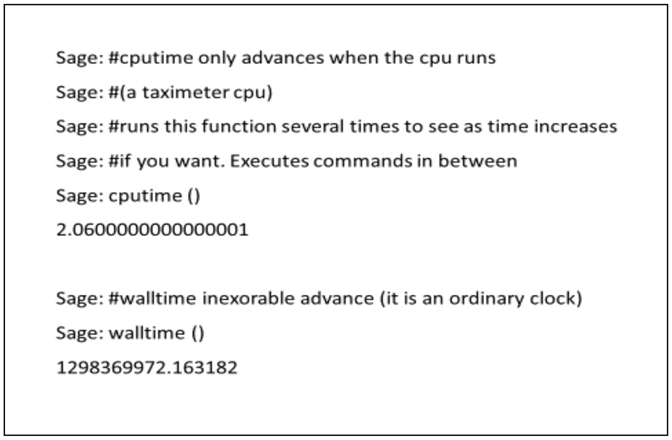


```
Sage: #To small data size
Sage. Time is_prime  (factorial (500) + 1)
False
Time: 0.09 s CPU, Wall: 0.009 s


Sage: #To higher data size, takes longer (in general)
Sage: time is_prime (factorial (100) + 1)
False
Time: 0.72 s CPU, Wall: 0.76 s
```

Fig. 12. Sage code to measure CPU time of ETL processes in both BI tools for small and higher data sizes

The time command is not flexible enough and needs the *CPUtime* functions and *Walltime*. *CPUtime* is a kind of meter: a meter progresses as the calculations are done, and moves many seconds as the CPU dedicated to Sage. The *Walltime* is a conventional clock (the clock UNIX). For the time spent on the program, also the before and after times of the execution were recorded and calculated and the differences are illustrated in Fig. 13.

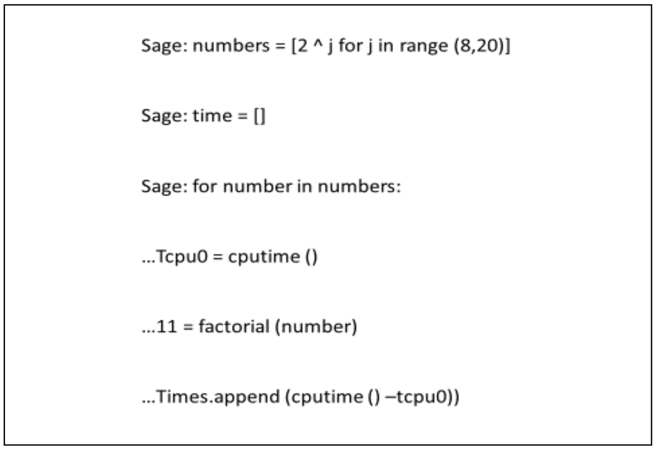


```
Sage: #cputime only advances when the cpu runs
Sage: #(a taximeter cpu)
Sage: #runs this function several times to see as time increases
Sage: #if you want. Executes commands in between
Sage: cputime ()
2.0600000000000001

Sage: #walltime inexorable advance (it is an ordinary clock)
Sage: walltime ()
1298369972.163182
```

Fig. 13. Sage code using *CPUtime* functions and *Walltime* to measure the ETL process in both tools

The following code saves the list of the CPU times used to run the factorial function with data of different sizes as shown in Fig. 14.



```
Sage: numbers = [2 ^ j for j in range (8,20)]

Sage: time = []

Sage: for number in numbers:

...Tcpu0 = cputime ()

...11 = factorial (number)

...Times.append (cputime () –tcpu0))
```

Fig. 14. Sage code to save lists of the CPU times used to run the factorial function with data of different sizes

B. Reporting Measurements

With Matlab the measured run time and efficiency of Reporting processes in both BI tools mentioned previously is shown in the flow chart in Fig. 15.

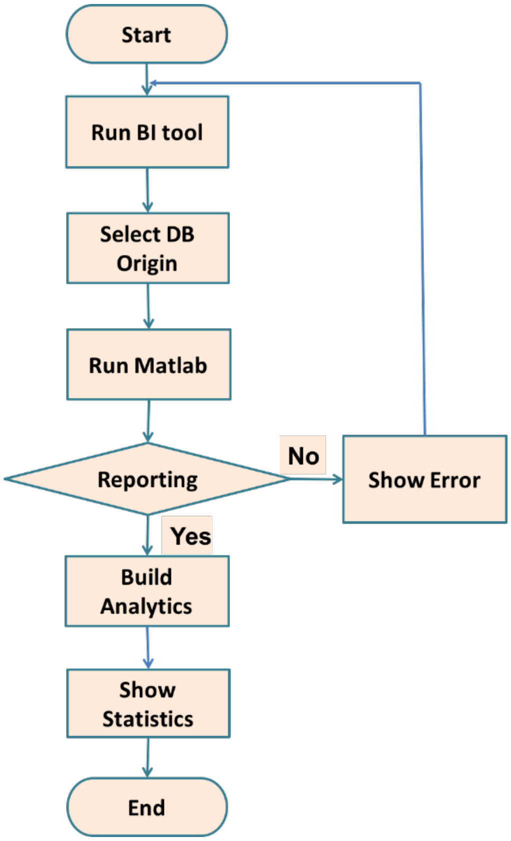
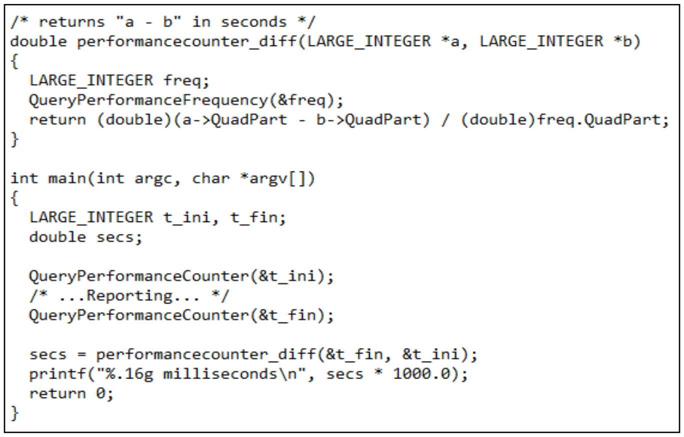


Fig. 15. The Reporting measurement process using the Matlab computer algebra system for all databases

For doing this activity, a C function was used that implemented a High-Resolution Performance Counter for measurement the Reporting processes on both BI tools as it is illustrated in Fig. 16.



```
/* returns "a - b" in seconds */
double performancecounter_diff(LARGE_INTEGER *a, LARGE_INTEGER *b)
{
  LARGE_INTEGER freq;
  QueryPerformanceFrequency(&freq);
  return (double)(a->QuadPart - b->QuadPart) / (double)freq.QuadPart;
}

int main(int argc, char *argv[])
{
  LARGE_INTEGER t_ini, t_fin;
  double secs;

  QueryPerformanceCounter(&t_ini);
  /* ...Reporting... */
  QueryPerformanceCounter(&t_fin);

  secs = performancecounter_diff(&t_fin, &t_ini);
  printf("%.16g milliseconds\n", secs * 1000.0);
  return 0;
}
```

Fig. 16. Code to measure the reporting process

In this case, Query Performance Counter acts as a clock () and Query Performance Frequency as CLOCKS_PER_SEC. That is the first function that gives the counter value and the second frequency (in cycles per second, hertz). It is clear that an LARGE_INTEGER is a way to represent a 64-bit integer by a union.

C. *Databases Analysis*

Six different Excel databases with different sizes have been used to perform the analysis. Those databases were acquired from UCI Machine Learning Repository [26] and their main features are described in Table 1:

TABLE I. DESCRIPTION OF THE SIX EXCEL DATABASES INCLUDING THEIR NUMBER OF ATTRIBUTES, INSTANCES AND SIZES

Database	***Number of Attributes***	***Number of Instances***	***Size***
DB 1	21	65.055	0.009 Mb
DB 2	26	118.439	0.017 Mb
DB 3	35	609.287	0.134 Mb
DB 4	40	999.231	1.321 Mb
DB 5	51	1.458.723	35.278 Mb
DB 6	62	2.686.655	144.195

D. *Computer system*

In order to perform the experiment and examination, the Business Intelligence Tools, Computer Algebra Systems and Databases are set on and customised in a PC with the following features: i) Operating system: x64-based PC, ii) Operating system version: 10.0.10240 N/D iii) Compilation 10240, iv) Number of processors: 1, v) Processor: Intel (R) Core (TM) i5-3317U vi) Processor speed: 1.7 GHz, vii) Instructions: CISC, viii) RAM: 12 Gb, ix) RAM speed: 1600 MHz, x) Cache: SSD express 24 Gb, xi) Math coprocessor: 80387, xii) GPU: HD 400 on board.

V. RESULTS

In this study, results for *CPUtime* as a function of the "size" was obtained from the data input for the ETL and Reporting processes from both Pentaho and Jaspersoft Business Intelligence Open Sources, applying two different Computer Algebra Systems.

The measurements of the computational times might fluctuate considerably based on many factors such as the used algorithm, operating system, processor speed, number of processors and instruction set that understand the amount of RAM, and cache, of each speed, along with math coprocessor, GPU among others. Even on the same machine, the same algorithm sometimes takes much longer to give the result of others, due to factors that it is more time-consuming than other applications that are running or if it has enough RAM when running the program.

Tables 2 and 3 shows the results of the CPU time (in minutes) of the ETL and Reporting processes and they present the times it took per tool in processing the different sized databases. Additionally, the increment of processing data can be considered as a difference between those BI tools in process. As a result of the first examination (Table 2), it is clear that the computational times for Pentaho ETL process measured by Sage were: 8 min; 12.01 min; 21 min; 32.01 min; 39.06 min and 48.01 min. Conversely, the computational times for Jaspersoft, were 9.54 min; 19.32 min; 31.88 min; 44.73 min; 55 min and 67.69 min, processing 0.009 Mb from DB1; 0.017 Mb from DB2; 0.134 Mb from DB3; 1.321 Mb from DB4; 35.278 Mb from DB5 and 144.195 Mb from DB6, respectively for both tools.

The results of the CPU time of the ETL process is shown in Table 2 and it presents the times that it took per tool in different databases.

TABLE II. RESULTS OF THE CPU TIME OF THE ETL PROCESS WITH THE TIMES TOOK PER TOOL AND THE INCREMENT IN THE PROCESS DATA IN THE DIFFERENT DATABASES

Tool	Process	Time (Minutes)					
		DB1	***DB2***	***DB3***	***DB4***	***DB5***	***DB6***
Pentaho	ETL	8	12.01	21	32.01	39.06	48.01
Jaspersoft	ETL	9.54	19.32	31.88	44.73	55	67.69
Increment in the Process of Data							
		DB1	***DB2***	***DB3***	***DB4***	***DB5***	***DB6***
Jaspersoft	ETL	19.22%	60.85 %	51.79 %	39.75 %	40.77 %	40.99 %

On the other hand, as a result of the second examination (Table 3), we can detect and see that the result of the Pentaho Reporting process measured by Matlab was: 3.75 min; 5.35 min; 8.47 min; 12.03 min; 17.07 min and 22.60 min. Conversely, the reporting process for Jaspersoft were 3 min; 4.02 min; 6.05 min; 8.13 min; 11.16 min and 14.15 min, processing 0.009 Mb from DB1; 0.017 Mb from DB2; 0.134 Mb from DB3; 1.321 Mb from DB4; 35.278 Mb from DB5 and 144.195 Mb from DB6, respectively for both tools. The results of the CPU time of the Reporting process are shown in Table 3 and it presents the time it took per tool in different databases.

TABLE III. RESULTS OF THE CPU TIME OF THE REPORTING PROCESS WITH THE TIMES TOOK PER TOOL AND THE INCREMENT IN THE PROCESS DATA IN THE DIFFERENT DATABASES

Tool	Process	Time (Minutes)					
		DB1	***DB2***	***DB3***	***DB4***	***DB5***	***DB6***
Pentaho	Reporting	3.75	5.35	8.47	12.03	17.07	22.60
Jaspersoft	Reporting	3	4.02	6.05	8.13	11.16	14.15
Increment in the Process of Data							
		DB1	***DB2***	***DB3***	***DB4***	***DB5***	***DB6***
Pentaho	Reporting	25%	32.99 %	40%	48%	53%	59.75 %

The Graphical comparison results of the CPU times for the ETL and Reporting processes performed by the BI tools, accessing six different sized databases which is illustrated below. In Fig. 17, we observe that Jaspersoft has significantly increased the results of the CPU time of the ETL process represented by 19.22%, 60.85% 51.79%, 39.75%, 40.77 and 40.99% processing DB1, DB2, DB3, DB4, DB5 and DB6, respectively. This means that in the ETL process, Pentaho had a better performance than Jaspersoft.

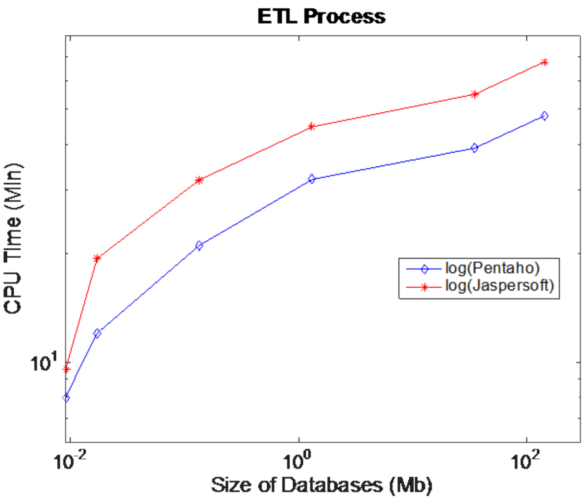


Fig. 17. CPU Time of the ETL process with the times it took per tool in processing data in the different databases

In Fig. 18, it is evident that Pentaho had a considerable rise in the outcomes of the Reporting process denoted by 25%, 32.99%, 40%, 48%, 53% and 59.75% processing DB1, DB2, DB3, DB4, DB5 and DB6, correspondingly. In this case, Jaspersoft had a better performance than Pentaho in the Reporting process.

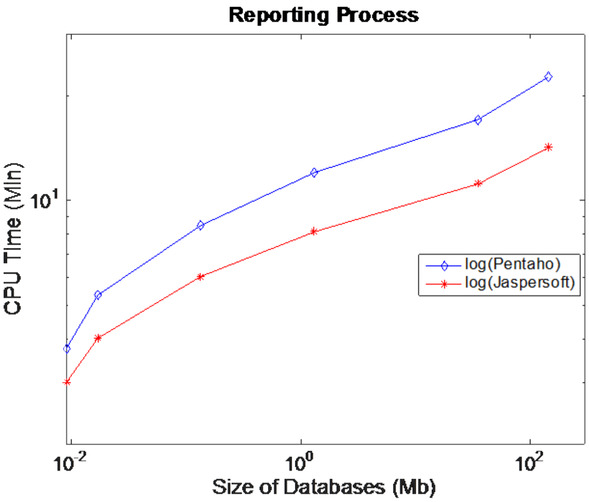


Fig. 18. CPU Time of the Reporting process with the times took per tool in the process data in different databases

The outcomes also showed that both results of CPU time of the ETL and Reporting process are directly related to the sizes of the databases. What is more, is that the study could identify that Pentaho had a superior performance for ETL process and Jaspersoft an improved performance for Reporting process.

VI. DISCUSSION

The ETL experimental analysis results clearly shows that Jaspersoft BI had an increment of CPU time in the process of data over Pentaho BI, represented in an average of 42.28% of performance metrics over six databases. Pentaho provided its data integration and ETL capabilities as shown in [9], having a better performance. Evidently, for this part of the experimentation, our study has demonstrated that Pentaho had higher performance ETL capabilities with the aim of covering the whole data integration requirements, simultaneously by big data as well. That high performance is provided by its parallel processing engine and these features are shown in [11].

Pentaho BI had a marked increment of the CPU time in the process of data over Jaspersoft evidenced by the Reporting analysis outcomes with an average 43.12% over six databases. Clearly, in this part of the examination, the analysis has confirmed that Jaspersoft has had a higher performance Reporting capability with the objective of generating reports. This particular feature is aligned with other studies, which argue that Jaspersoft extends the range of its BI requirements including reporting based on its operational production, interactive end-user query, data integration and analysis as shown in [11]. On top of this, investigating various security features [27-29] could be an interesting avenue to explore in the future to protect BigData.

VII. Conclusion

This study has tested two of the best positioned open source Business Intelligence (BI) systems in the market: Pentaho and Jaspersoft. Both BI systems present notable features on their components. **Pentaho** on one side along with ETL component with great usability, maintainability and flexibility in making the transformations: Web Application with Java j2EE application 100% extensible, adaptable and configurable; the configuration management is integrated in most environments, that communicate with other applications via web services; it integrates all the information resources into a single operating platform; Reports with an intuitive tool that allows clients to create reports easily; OLAP Mondrian with a consolidated engine widely used in environments of JAVA; Dashboard Designer makes dashboards Ad-hoc, dashboards based on SQL queries or Metadata and a great freedom by offering a wide range of components and options. **Jaspersoft** on the other side has JasperETL (Talend) with Java / Perl native, Web Application with a Java j2EE application 100% extensible, adaptable and customizable; the management settings are very well resolved, it allows almost all through the same Web application; It integrates all information resources into a single operating platform; the editor Ad-hoc reports and Box Editor Ad-hoc command are best resolved; Reports are fast; Ad hoc and have a nice interface, with good flexibility and power, simple, intuitive and easy to use.

The experimental analysis has focussed on their ETL and Reporting processes by measuring their performance s using the two Computer Algebra Systems, Sage and Matlab. During the ETL analysis results, clearly showed that it could observe Jaspersoft BI and has an increment of CPU time in the process of data over Pentaho BI, represented in an average of 42.28% of performance metrics over six databases. Meanwhile, Pentaho BI had a marked increment CPU time in the process of data over Jaspersoft evidenced by the Reporting analysis outcomes with an average 43.12% over the databases. This study is a useful reference for many researchers and those who are supporting decisions of Big Data processing and the implementation of BI open source tool based on their process expectations. The future work of the author would involve new studies and implementations of BI with Data warehousing to create a technological tool to support the decision-making at the enterprise level by taking this paper as a base.

References

[1] B. List, R. M. Bruckner, K. Machaczek, J.Schiefer, “A Comparison of Data Warehouse Development Methodologies Case Study of the Process Warehouse,” in Database and Expert Systems Applications - DEXA 2002, France, 2002.

[2] H. Dresner, “Business intelligence: competing Against Time.,” in Twelfth Annual Office Information System Conference, London, 1993.

[3] S. Atre, L. T. Moss, “Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, Addison Wesley Professional”, 2003.

[4] J. F. Gonzalez, “Critical Success Factors of a Business Intelligence Project,” Novática, no. 211, pp. 20-25, 2011.

[5] R. L. Sallam, B. Hostmann, K. Schegel, J. Tapadinhas, J. Parenteau, T. W. Oestreich, “Magic Quadrant for Business Intelligence and Analytics Platforms”, 23 February 2015. [Online]. Available: www.gartner.com/doc/2989518/magic-quadrant-business-intelligence-analytics. [Accessed 9 Aug 2016].

[6] Gartner, Inc., “IT Glossary”, 2016. [Online]. Available: http://www.gartner.com/it-glossary/business-intelligence-bi/. [Accessed 9 Aug 2016].

[7] R. Kune, P. K. Konugurthi, A. Agarwal, R. R. Chillarige, R. Buyya, “The Anatomy of Big Data Computing”, Software: Practice and Experience, pp. 79-105, 2016.

[8] Pentaho A Hitachi Group Company, “Pentaho | Data Integration, Business Analytics and Bid Data Leaders”, Pentaho Corporation, 2005-2016. [Online]. Available: www.pentaho.com. [Accessed 10 Aug 2016].

[9] D. Tarnaveanu, “Pentaho Business Analytics: a Business Intelligence Open Source Alternative”, Database System Journal, vol. III, nº 3/2012, p. 13, 2012.

[10] T. Kapila, “ Pentaho BI & Integration with a Custom Java Web Application”, 2014. [Online]. Available: www.neevtech.com/blog/2014/08/13/pentaho-bi-integration-with-a-custom-java-web-application-2/. [Accessed 11 Aug 2016]

[11] Innovent Solutions, “Pentaho Reports Review ”, 2016. [Online]. Available: www.innoventsolutions.com/pentaho-review.html. [Accessed: 12 Aug 2016].

[12] G. Pozzani, “OLAP Solutions using Pentaho Analysis Services”, 2014. [Online]. Available: www.profs.sci.univr.it/~pozzani/attachments/pentaho_lect4.pdf. [Accessed: 12 Aug 2016].

[13] Sanket, “Fusion Charts Integration in Pentaho BI Dashboards”, 2015. [Online]. Available: www.fusioncharts.com/blog/2011/05/free-plugin-integrate-fusioncharts-in-pentaho-bi-dashboards/. [Accessed: 13 Aug 2016].

[14] TIBCO Jaspersoft, “Jaspersoft Business Intelligence Software”, TIBCO Software, 2016. [Online]. Available: www.jaspersoft.com. [Accessed 15 Aug 2016].

[15] S. Vidhya, S. Sarumathi, N. Shanthi, “Comparative Analysis of Diverse Collection of Big data Analytics Tools”, International journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 8, nº 9, p. 7, 2014.

[16] T. olavsrud, “Jaspersoft Aims to Simplify Embedding Analytics and Visualizations”, 2014. [Online]. Available: www.cio.com/article/2375611/business-intelligence/jaspersoft-aims-to-simplify-embedding-analytics-and-visualizations.html. [Accessed: 16 Aug 2016]

[17] S. Pochampalli, “Jaspersoft BI Suite Tutorials”, 2014. [Online]. Available:www.jasper-bi-suite.blogspot.com.au/. [Accessed: 17 Aug 2016].

[18] J. Vinay, “OLAP Cubes in Jasper Server”, 2013. [Online]. Available: www.hadoopheadon.blogspot.com.au/2013/07/setting-up-olap-cubes-in-jasper.html. [Accessed: 19 Aug 2016.]

[19] Informatica Data Quality Unit, “Data Quality: Dashboards and Reporting”, 2013. [Online]. Available:www. Markerplace.informatica.com/solution/data_quality_dashBoards_andreporting-961. [Accessed: 21 Aug 2016]

[20] Sagemath, “Sagemath | Open-Source Mathematical Software System”, Sage, 20 March 2016. [Online]. Available: www.sagemath.org. [Accessed 21 Aug 2016].

[21] AIMS Team, “Sage”, 2016. [Online]. Available: www.launchpad.net/~aims/+archive/ubuntu/sagemath. [Accessed: 21 Aug 2016.]

[22] W. Stein, “The Origins of SageMath”, 2016. [Online]. Available: www.wstein.org/talks/2016-06-sage-bp/bp.pdf. [Accessed: 28 2016.]

[23] MathWorks, “MATLAB - MathWorks - MathWorks Australia”, MathWorks, 3 March 2016. [Online]. Available: www.au.mathworks.com. [Accessed 28 Aug 2016].

[24] M. S. Gockenbach, “A Practical Introduction to Matlab”, 1999. [Online]. Available: www.math.mtu.edu/~msgocken/intro/intro.html. [Accessed: 28 Aug 2016]

[25] K. Black, “Matlab Tutorials”, 2016. [Online]. Available: www.cyclismo.org/tutorial/matlab/. [Accessed: 29 Aug 2016].

[26] M. Lichman, “UCI Machine Learning Repository”, 2013. [Online]. Available: www.archive.ics.uci.edu/ml [Accessed 10 Aug 2016].

[27] D. V. Pham, A. Syed, A. Mohammad and M. N. Halgamuge, “Threat Analysis of Portable Hack Tools from USB Storage Devices and Protection Solutions", International Conference on Information and Emerging Technologies, pp 1-5, Karachi, Pakistan, 14-16 June 2010.

[28] D. V. Pham, A. Syed and M. N. Halgamuge, “Universal serial bus based software attacks and protection solutions, Digital Investigation 7, 3, pp 172-184, 2011.

[29] D. V. Pham, M. N. Halgamuge, A. Syed and P. Mendis. “Optimizing windows security features to block malware and hack tools on USB storage devices”, Progress in electromagnetics research symposium, 350-355, 2010.