Open Source BI Platforms: a Functional and Architectural Comparison

Matteo Golfarelli

DEIS, University of Bologna, Viale Risorgimento 2, Bologna, Italy
matteo.golfarelli@unibo.it

Abstract. While in the past the BI market was strictly dominated by closed source and commercial tools, the last few years were characterized by the birth of open source solutions: first as single BI tools, and later as complete BI platforms. An Open Source BI platform provides a full spectrum of BI capabilities within a unified system that reduces the overhead for the development and management of each application, and lets the user feel like he/she was using a single BI solution. This paper proposes a comparative evaluation of three different Open Source BI platforms (namely JasperSoft, Pentaho and SpagoBI) aimed at understanding their current features, their future potentialities and their limits when adopted in real projects as well as a basis for research prototyping. Overall we try to understand if the open source phenomenon will be able to become a valid alternative to commercial platforms within the BI context.

1 Introduction

While in the past the BI market was strictly dominated by closed source and commercial tools (see for example [1] for different vendors' market shares), the last few years were characterized by the birth of open source (OS) solutions. At first OS BI tools covered isolated portions of the DW process with a limited set of functionalities that made them appear as toys if compared to large commercial BI platforms. Consider for example the initial releases for Octopus as to ETL, Mondrian as to OLAP servers, and JPivot as to OLAP clients (see [2] for a complete listing). While single tools still keep evolving with an increasing number of features and a higher level of reliability, the turning point in OS BI was the birth of OS BI platforms. An *OS BI platform* provides a full spectrum of BI capabilities within a unified system that reduces the overhead for the development and management of each application, and lets the user feel like he/she was using a single BI solution.

Commercial platforms are commonly considered superior to OS ones. Nevertheless, we believe that OS BI platforms will evolve much faster than commercial ones since they are not constrained by compatibility problems and rigid (or even obsolete) architectures. Furthermore, OS solutions can exploit the contributions of the OS development community, that relies on hundreds of programmers and designers as well as on the direct involvement of researchers.

This paper presents a comparative evaluation of three different OS BI platforms (namely JasperSoft, Pentaho and SpagoBI) aimed at understanding their current features, their future potentialities and their limits when adopted in real projects.

Overall we try to understand if the open source phenomenon will be able to become a valid alternative to commercial platforms within the BI context. OS BI platforms are not only attracting practitioners but also researchers since the availability of the source code makes them a perfect framework for prototyping and testing research findings. Furthermore both the European Community [3] and the United States government, as well as many other countries [4] are urging for the adoption of open source solutions in their research programs and more in general in the ICT area as a lever for increasing competitiveness [5]. Nowadays, in several areas such as e-health and e-government, funding calls suggest (or occasionally require) the use of open source.

The diffusion of OS BI technologies is also supported by private companies and consortiums. For example, BI Initiative [6] is an interesting OW2 project aimed at the diffusion of OS BI technologies. In particular BI Initiative is aimed at improving the coordination effort in the OS BI context, increasing the use of OS BI solutions at enterprise level, strengthening connections between integrators, vendors, users and the research communities and finally attracting more attention from the research activities to foster innovative BI solutions and practices.

The only scientific paper focusing on OS BI is the one proposed by Thomsen and Pedersen [2]: this interesting survey focuses on functionalities available in single tools but it does not consider BI platforms. A large number of comparative analyses are periodically published by software vendors, that obviously report a biased point of view, as well as independent groups. These reports (see for example [7,8]) are typically tailored on practitioners' needs and focus on technical aspects rather than studying the overall characteristics of the suite. The quality of OS software has been studied in three projects funded by the European Union, namely Flossmetric – Free/Libre Open Source Software Metrics - [9], Qualoss - Quality in Open Source Software - [10], and SQOOS - Software Quality Observatory for Open Source Software - [11]. The three projects converged to a unique initiative, named flossquality, aimed at developing a high level methodology to benchmark the quality of OS software and to apply it to a large number of OS projects. None of the platforms considered have currently been analyzed.

Our paper is thus the first one studying the added value of OS BI platforms; it evaluates comparatively the philosophy of the different platforms as well as their architecture, functionalities and usability. We will not consider efficiency aspects since they are strictly determined by the single BI tools which are often shared by the different platforms.

The paper is structured as follows. Section 2 briefly describes how the comparison has been conducted and introduces the key aspects that have been analyzed. Section 3 describes the platforms from different points of view, while in section 4 the results of the comparison are reported and discussed.

2 Method of conducting the comparison

This work comes from the interest in exploring the OS BI platforms shared by our research group and three Italian consulting firms that intend to propose OS based

applications to their customers. The outcome of the analysis is the fusion of our independent analysis and their work on field testing and evaluation. All the consulting firms[1] involved are specialized in BI projects and they usually develop their applications using commercial BI suites.

We initially defined an evaluation grid describing in details the aspects to be investigated. The evaluation criteria were derived from the models available in the literature for general purpose software [12,13] and they were specialized to fit BI software specificities. The resulting grid was shared with the consultant firms and further discussed and integrated. Each consultant firm carried out one or more porting of real projects previously implemented through commercial BI suites. The compiled grids were finally shared and discussed with the other participants. In the current work we only report and summarize the evaluations concerning the platforms while we do not study in depth the features of each single BItool. The comparison hinges on the following key aspects:

- Non-technical: platform philosophy, type of licensing and availability of enterprise editions.
- Architectural: in terms of the global framework, modules and their relationships, programming languages and supported operational systems.
- Functional: in terms of functionalities provided natively by the platforms or made available to the users through the integrated BI tools.
- Meta-data: in terms of expressiveness, completeness, standardization and level of reusability.
- Security: in terms of functionalities provided for authentication and profiling of the users, interfaces to external authentication systems and secure data transmission.
- Usability: both from the user viewpoint, in terms of level of transparency in using the different tools, and from the developers' and system administrators' viewpoint in terms of complexity of installation and administration as well as development of applications, quality of manuals and forums.

3 Platforms description

The platforms we considered are JasperSoft BI Suite [14], Pentaho BI Suite [15] and SpagoBI [16] and the versions considered are those released by December 31 2008. In the following will refer to them with the names Jasper, Pentaho and SpagoBI, respectively. Please note that in many cases there is a gap between the functionalities that are actually available to the users and those expected by the project road map for a given release. We will adopt a strict policy and we will disregard those features that have been only sketched.

[1] We do not report the company names since they required to remain anonymous in order to avoid marketing activities by both open source and commercial software producers.

3.1 Non-technical aspects

The three platforms adopted two different open source models:

- Commercial open source: this model provides for an open source product that meets the user's basic needs (i.e. community edition); an enterprise edition of the product can be purchased and it usually includes enhanced features as well as support and training services. Jasper and Pentaho fit into this model. Their community editions are covered by the GNU General Public License (GPL) and Mozilla Public Licence (MPL) respectively while commercial agreements are needed for the enterprise releases.
- Free and Open Source Software (FOSS): the product is completely free, no enterprise solution is available, thus all the functionalities are available to the community for free. SpagoBI fits into this model. It is distributed under the GNU LGPL license.

Without entering into details, the right to freely use, modify, and redistribute software is fundamental to the GNU GPL agreements [17] and if you release a modified version of a software, you may be obliged to contribute your entire work to the open source community under the same type of agreement. On the other hand, a commercial agreement typically allows you to use but not to distribute the software. Usually, a different type of agreement (OEM license) is needed for profit developers who want to include BI capabilities in their applications.

According to the OS philosophy, platform functionalities can either be developed internally by the software house that owns the platform (e.g. JasperReport was born within JasperSoft Corporation) or, more frequently, they can be achieved by plugging a module implemented in a different OS project. Module plugging can be obtained by:

- Integration: a software interface is defined in order to control and to exploit module functionalities directly and transparently through the platform. The intellectual property of the software does not change, and the original developers remain in charge of maintaining and evolving the module.
- Acquisition: the intellectual property of the software is acquired and the original project terminated. The buyer will be in charge of maintaining and evolving the module.
- Technological partnership: stands in the middle between integration and acquisition. The original project remains alive and it is maintained by the original developers. The partner that incorporates the module influences its evolution and collaborates to its maintenance. The module usually appears with a different name in the new platform.

The policy adopted changes depending on the complexity of modules and on its relevance to the platform. Pentaho often has recourse to acquisition (e.g. Pentaho ETL comes from the Kettle project) while SpagoBI is strictly based on integration; finally Jasper mainly exploits partnerships (e.g. JasperETL was developed through a partnership with Talend that still maintains Talend Open Studio that is also integrated in SpagoBI).

Platforms that acquire the BI engines or have strong partnerships with their original developers can steer and control the engine evolutions and ensure a higher level of quality; on the other hand, platforms that integrate third-party modules can lean on wider developer communities and can more easily include new BI projects.

3.2 Architectural aspects

An OS BI platform provides a full spectrum of BI capabilities within a unified system that reduces the overhead for the development and management of each application, and lets the user feel like he/she was using a single BI solution.

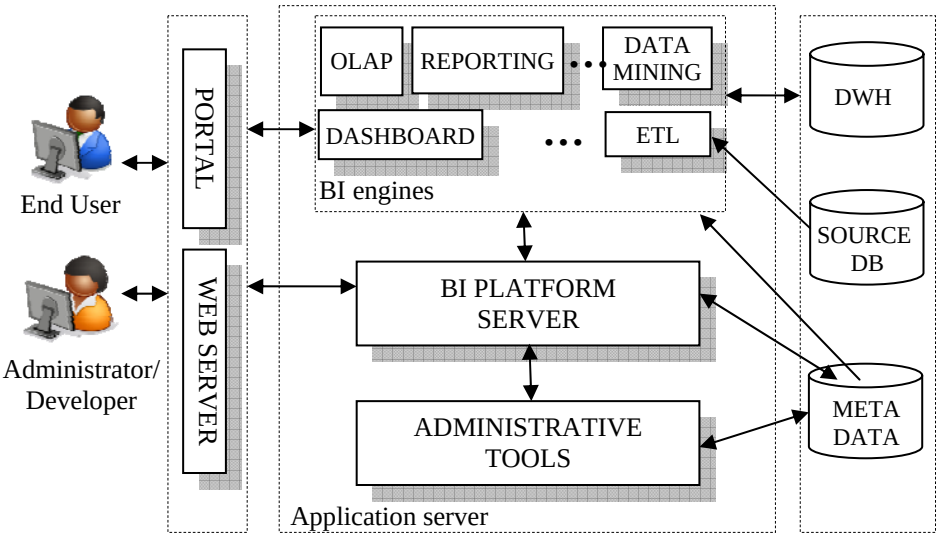

Figure 1. Reference architecture for BI OS platforms. Arrows entering group of modules mean that communication concerns all the modules.

OS BI platforms are developed using Java since the modules they rely on are based on this technology. They typically require an application server and the users, as well as system administrators and developers, access them through a web browser. The platforms adopted the same architecture that is sketched in Figure 1: the platform core is a web application that stands in the middle between BI engines that implement each single BI functionality and the databases that store the required information. The users access the system through a web client that can be connected either to a portal or directly to a web server. A meta knowledge layer completes the picture and is crucial to provide the platforms with the necessary “intelligence”. A typical user-platform interaction includes the following steps: (1) the user requiring a given document logs into the portal or directly into the platform server; (2) the platform server verifies if the user profile allows him/her to access the document requested; (3) the platform server opens the connection to the data source; (4) the platform server also activates

the BI engine involved and passes it the user credentials, the necessary meta-information as well as the connection to the data source; (5) the BI engine produces the document and makes it available to the user through the web server or the portal.

Table 1: Modules building up the considered BI OS platforms, alternative configurations are possible.

Modules	JasperSoft	Pentaho	SpagoBI
Application Server	JBoss	JBoss	JBoss
Authentication and user profiling	Acegi	Acegi	Integrated in eXo Portal
Collaboration	-	-	Dossier
Dashboard	JFreeChart	JFreeChart	Openlaszlo
Data Mining	-	Weka	Weka
DBMS	MySQL, Oracle, SQL Server, PostgreSQL, etc.	MySQL, Oracle, SQL Server, PostgreSQL, etc.	MySQL, Oracle, SQL Server, PostgreSQL, etc.
ETL	JasperETL	Pentaho Data Integration	Talend Open Studio
Geo-referencing	Google Maps	Google Maps	GEO
Job Scheduler	Quartz	Quartz	Quartz
OLAP	Mondrian&Jpivot	Mondrian&Jpivot	Mondrian&Jpivot
Portal	Liferay	JBoss Portal	ExoPortal, Liferay
Query by Example	-	-	Hibernate
Reporting	JasperReport	Pentaho Report Designer, JasperReport, BIRT	JasperReport, BIRT
Single sign on	Acegi	CAS	CAS
Web Server	Tomcat	Tomcat	Tomcat

Beside front-end functionalities the platforms include back-end ones as for example ETL services and scheduling services necessary to automate report updates. In all these cases engines are activated directly by the platform or by the system administrator.

Table 1 shows the main modules building up the platforms considered. Many of the modules are shared, some of them are evolutions of a different open source project

(e.g. Jasper ETL comes from Talend Open Studio), others have been developed internally and belong to the same software house that is charge of the platform (JasperReport is the most widespread modules for BI reporting, while GEO is the module developed by SpagoBI team for geo-referenced analysis) - reusing and sharing underlie OS software development. Table 1 also shows that some modules are standard de facto within BI OS: in particular the Mondrian OLAP engine and the JPivot graphical interface are the standard solutions for OLAP, while Weka is the standard data mining module.

3.4 Metadata

Within a BI platform, metadata largely determine the behavior it can exhibit, and the expressivity of reports and OLAP analyses. Metadata store the structure of data sources and multidimensional cubes, the content of the reports and the actions to be executed within an ETL process. Metadata also store user profiles as well as information related to scheduling and auditing.

We distinguish between platform metadata and BI engine metadata. In fact, metadata necessary to specific BI functionalities are usually created outside the platforms by editing an XML file or by exploiting simple graphical tools. Only afterwards can they be imported in the BI platform. Although, they model the same information, metadata belonging to different engines are differently coded and cannot be reused. This obviously affects development and maintenance negatively. For example, the multidimensional structure of a cube must be defined repeatedly if the cube is involved in an OLAP analysis, in a report or in a ETL process. We believe that this is the main shortcoming of OS BI platforms compared to commercial ones that are typically based on a unique and integrated metadata repository. Within community editions metadata are stored in XML files, while the two enterprise editions provide for a DBMS based metadata repository. Although all three platforms declare that their metadata are CWM-compliant [18] no interoperability tools have been released yet.

3.3 Functional Aspects

Table 2 reports the main functionalities made available by the platforms. If we consider the completely free version of the suites (i.e. community editions) SpagoBI overcomes Pentaho and Jasper that make available many of the advanced features only in the enterprise editions. We will not discuss in detail each single item in the table since most of them are self explaining, we will briefly describe the infrequent terms instead. The term *Query by Example* refers to the capability of running free inquiring over a database schema using a graphical interface that does not require the user to be an SQL expert, while *Ad-hoc reporting* refers to the availability of a graphical interface that allows each user to create his own reports directly from a web interface. The term *collaborative BI* refers to functionalities that allow BI results to be shared between managers in order to reach a concerted decision. Finally, *report*

validation workflow stands for the possibility of defining a set of states and approval steps a report and its data must pass through before being finally published.

Table 2: Main functionalities made available by the platforms; community and enterprise releases are distinguished.

Functionalities	SpagoBI	Pentaho	Pentaho Ent. Ed.	Jasper	Jasper Ent. Ed.
Activities scheduling	√	×	√	×	√
Ad-hoc reporting	×	×	√	×	√
Auditing	√	×	√	√	√
Collaborative BI	√	×	×	×	×
Data Mining	√	√	√	×	×
Dashboard	√	√	√	×	√
Document export	√	√	√	√	√
ETL	√	√	√	√	√
Geo-referenced analysis	√	√	√	×	√
OLAP	√	√	√	√	√
Query by Example	√	×	×	×	×
Report validation workflow	√	×	√	×	×
Reporting	√	√	√	√	√
User profiling	√	×	√	×	√

Security issues are particularly relevant in data warehousing. All the platforms allow secure data transmission as well as user authentication, while they offer pretty different functionalities for user profiling. Typically DBMSs are not suitable for defining the security policies relevant in a BI application, thus BI platforms are in charge of their definition. In advanced commercial solutions profiling is based on security models that govern, in a centralized fashion, three fundamental areas of every BI application: (1) *objects* (e.g. a specific report or an OLAP analysis) each user can use, (2) *cell-level data* each user can access; and (3) *BI functionalities* each user gets (i.e. choosing the types of actions users may perform in the system such as printing, saving, exporting, drilling, pivoting, sorting, formatting and creating reports). As to OS BI platforms only two of these areas are covered; in fact user profiles can grant or deny access to different objects and allow filters to be applied on data retrieval but they cannot restrict the set of BI functionalities a user can run on a given document. More in detail user profiling is made available for free by SpagoBI while Jasper and Pentaho offer this feature only in their enterprise editions: Pentaho community edition only provides user authentication, while Jasper community edition provides a simplified profiling where the access is granted/denied for an entire directory usually containing mode reports or analysis.

As concerns the comparison between community - including SpagoBI - and enterprise editions, differences are not only in terms of functionalities available to the users (see Table 2) but also in terms of utilities for administrators and developers. The main improvements we identified in enterprise editions are:

- Improved administration consoles: the improvement is particularly relevant in Pentaho where the Enterprise console fills the gap with Jasper as concerns usability and functionalities.
- Wizard based configurations: most configuration activities are based on wizards and do not require a manual access to configuration files or multiple access to menus.
- Process monitoring: front-end (e.g. query execution) as well as back-end (e.g. ETL) processes can be monitored and analyzed in order to optimize their execution.
- ETL debugging environment: it is available and determines a strong reduction of the development effort.

Administrators and developers are further supported through a wider documentation, a knowledge base as well as consultant and training services. Obviously such enhancements, together with warranties and certification of the software on a larger number of operating systems, applications servers, DBMSs, etc., become more and more relevant when you are developing a mission-critical application or when you are planning to adopt the platform in a large and complex organization.

3.5 Usability

Usability enables the users to easily access BI functionalities and it ensures developers and administrators a high productivity.
From the user point of view platforms usability is largely determined by the BI engines composing them. We consider the usability of those engines qualitatively satisfactory. Although they do not reach the level of refinement of the commercial suites, their graphical features give the developed applications an appreciable look-and-feel. OS BI platforms also succeed in hiding the access to different tools.
From the administrators' point of view usability is determined by the easiness in administering the platform and adding new functionalities, in particular:

- Complexity of the installing and configuring process: installing procedures are in general quite easy. This is particularly true for Pentaho and JasperSoft whose installation procedures completely rely on a wizard that also includes the installation of the BI engines. SpagoBI installation requires manually modifying eXoPortal configuration files and it does not include BI engines that must be installed separately.
- Administration complexity: the different usability is well perceived when you register a new report or analysis. As described in Section 3.2 functionalities (e.g. a report, an OLAP analysis, an ETL process) are usually developed outside the platform and then imported before making them available. In SpagoBI and even

more in Jasper we appreciated the easiness of the form-based procedure. Much effort is needed in Pentaho where functionalities registration is based on Action Sequences: an Eclipse procedure that may become quite complex since it is not adequately supported by appropriate debug information and documentation. This problem is partially solved in the enterprise edition that includes a debug tool for Action Sequences.

- Problem solving and training effort: manuals have a good quality and they allow most of the problems to be solved. Besides, in line with OS philosophy, several practitioners' forums make available a high number of technical tips. The quality of information and the activeness of the forums are strictly related to the number of the platform users. During our analysis, the richness and most active forum was the one from Pentaho (more than 20,000 registered users). The Jasper community is even larger (about 90,000 registered users) but we experienced in many cases longer response time (about 2-3 days for receiving an answer). SpagoBI community is definitely smaller and so the activeness of its forum (the number of registered users is unavailable, but only six thousands posts have been submitted since 2006). Finally, the adoption of standard and well-known programming languages does not require programmers and administrators to have any particular skill.

5 Discussion and Conclusions

Our analysis shows that OS BI platforms determine an added value with respect to single BI tools since they allow several functionalities to be accessed transparently and a set of processes to be centralized and simplified thus reducing the administration and development effort. We believe that the main shortcoming of the platform is the absence of a fully centralized and unified metadata layer, as this reduces reusability and integration. The capabilities of the administrative tools could also be improved in the community editions – this concerns in particular Pentaho.

SpagoBI makes available a remarkable number of BI functionalities even if it adopts a free open source model. As concerns the functionalities offered to the users SpagoBI is comparable to the enterprise editions by Jasper and Pentaho. From that observation we can infer that integration (instead of acquisition) allows an easier plug of new modules and gives the original developers the possibility to improve them. On the other hand, acquisition ensures a higher quality of the modules and a road map compatible with the owner's one. These are mandatory needs for distributing certified editions.

Although OS BI platforms are still not as sophisticated as commercial ones we can state that they got a sufficient level of reliability and must be considered a valid alternative to commercial suites. This is particularly true in small and medium-sized enterprises where the quantity of data and the workload are not critical points. Several companies are evaluating the use of OS BI in pilot projects where budget constraints are typically very tight. The main risks related to an investment in OS technology come from unexpected termination of the project that will no longer be maintained and evolved or, even worse, from the adoption of a more restrictive licensing of the

new releases that prevents using or distributing them. Finally, due to the short history of such products, it is impossible to predict if, apart from the initial investment, the companies that are in charge of the platforms will earn enough from services and application developments to stay on the market.

According to their road maps and evolution trends OS BI platforms will equal commercial ones in a few years. In order to really do better than commercial solutions, we argue, OS BI platforms should not only replicate commercial functionalities with lower costs for the final users, but should also propose innovative functionalities according to the most sophisticated requirements of business users. Coupling twenty years of experience in building BI software with the more recent results on BI research can really make the difference.

References

[1] OLAP Report: OLAP Market Share Analysis Retrieved March 12, 2009 from The Olap Report web site: http://www.olapreport.com/market.htm.
[2] Thomsen, C., Pedersen, T. B.: A Survey of Open Source Tools for Business Intelligence . In DaWaK, pp. 74–84, 2005.
[3] DG Information Society and Media: Towards a European Software Strategy – Report of an Industry Expert Group. Retrieved March 12, 2009 from The European Commission web site: http://www.nessi-europe.com/
[4] Lewis, J. A.: Government Open Source Policies. Center for Strategic and International Studies, 2007.
[5] Ghosh, R.A.: Study on the: Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU. Retrieved March 12, 2009 from The European Commission web site: http://ec.europa.eu/enterprise/ict/studies/publications_en.htm.
[6] BI Initiative web site http://www.ow2.org/view/BusinessIntelligence/
[7] Optaros. Open Source Catalog, Optaros white paper, 2009.
[8] Smile. Décisionnel Solutions open source, 2009.
[9] Flossmetric Project web site: http://www.flossmetrics.org/
[10] Qualoss Project web site: http://www.flossmetrics.org/
[11] SQOOS Project web site: http://www.sqo-oss.eu/
[12]Samoladas I., Gousios G., Spinellis D., Stamelos I.: The SQO-OSS Quality Model: Measurement Based Open Source Software Evaluation. Open Source Development, Communities and Quality, vol. 275, pp. 237-248, 2008.
[13] Ortega M., Perez M., Royas, T. Construction of a Systemic Quality Model for Evaluating a Software Product, Software Quality Journal, 11, 219–242, 2003.
[14] JasperSoft web site http://www.jaspersoft.com/
[15] Pentaho web site http://www.pentaho.com/
[16] SpagoBI web site http://spagobi.eng.it/
[17] GNU General Public License web site http://www.gnu.org/copyleft/gpl.html
[18] CWM: Common Warehouse Metamodel Specification - Version 1.1, OMG Inc., 2003.