

SpagoBI Overview

Author

Grazia Cazzin
Gabriele Ruffatti (review)

Index

| | |
|-------------------------------|---|
| VERSION | 3 |
| DOCUMENT GOAL..... | 3 |
| REFERENCES | 3 |
| 1 SPAGOBİ GOALS | 4 |
| 2 SPAGOBİ IN A NUTSHELL | 4 |
| 3 SPAGOBİ FEATURES | 5 |
| 4 SPAGOBİ ARCHITECTURE..... | 6 |

Version

| | | | |
|-----------------------------|--|-------------------------------|-----------------|
| Version/Release n° : | 1.3 | Data Version/Release : | June, 20th 2005 |
| Update description: | Architectural schema update | | |
| Version/Release n° : | 1.2 | Data Version/Release : | June, 17th 2005 |
| Update description: | Change of SpagoBI license | | |
| Version/Release n° : | 1.1 | Data Version/Release : | Maj, 20th 2005 |
| Update description: | Architectural schema update References update | | |
| Version/Release n° : | 1.0 | Data Version/Release : | April, 7th 2005 |
| Update description: | First release | | |

Document goal

This document provides a first description of the relevant aspects of SpagoBI platform, with particular evidence of its architectural aspects.

References

Some of the concepts of this document refer to the following documentation:

- [1] BERRY, M. J. A., LINOFF, G. A. *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management*, Wiley Publishing, New York 2004².
- [2] HALPIN, T., *Information Modeling and Relational Databases. Form conceptual analysis to logical design*, Morgan Kauffman Publisher, San Francisco 2001.
- [3] KIMBALL, R., ROSS, M., *The data warehouse toolkit*, Wiley, New York 2000².
- [4] KIMBALL, R., CASERTA, J., *The Data WarehouseETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*, Wiley Publishing, New York 2004.
- [5] MARCO, D., *Building and Managing the Meta Data Repository: A Full Lifecycle Guide*, Wiley Publishing, New York 2000.
- [6] RICHARDSON, W.C., AVONDOLIO, D., VITALE, J., LEN, P., SMITH, K.T., *Professional Development With Open Source Tools*, Wrox, Indianapolis 2004.
- [7] WITTEN, I.H., FRANK, E., *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kauffman Publisher, 2000.

Information about the Spago framework are available on http://spago.eng.it/docs_en/documentation/index.html

1 SpagoBI goals

The purpose of SpagoBI project is to realize a strong and organic infrastructure for the development of Business Intelligence (BI) solutions.

At present there are many free open source (FOSS) specific products realizing single particular BI aspects: reporting tools, Data Mining instruments and algorithms, EAI products and so on. Each of these products is useful in its specific domain but when a complete solution is needed they all put together do not give the real answer. Actually only big commercial solutions may cover several Business Intelligence's aspects.

SpagoBI will use the most interesting solutions (the best of the breed FOSS solutions) and new software components in order to realize a really integrated Free Platform. The project is ambitious and very complex as well. It doesn't release a package for the community but it starts as a development project going to involve a community interested on it. For this aim, the Engineering Ingegneria Informatica development team will release the architectural and technical design documentation and, in the meanwhile, starts the development of the core platform, waiting for the participation of other workgroups.

Every component will be released as GNU LGPL licence in order to share, develop, complete, extend and improve the final platform, according to the cooperative behaviour of the free software communities.

Spago BI will grow in time according to the development road map and on time will be revised according to the community's suggestions.

SpagoBI will be supported in terms of professional services and maintenance by Engineering Ingegneria Informatica.

2 SpagoBI in a nutshell

SpagoBI is a Business Intelligence Free Platform, completely realized according to the free software philosophy. It is a suite of coordinated and integrated tools allowing the development of a specific BI solution in every business area and market segment. SpagoBI has a modular structure integrated into the core system. It guarantees solidity and harmony to the platform and a great extension capability.

Not all the SpagoBI modules are always necessary for every project: you can use the appropriate SpagoBI module for your project. You can start using a single module with the guarantee that further extensions will be easy because everything is inserted in an overall vision.

SpagoBI uses many technologies and products already available as Free Open Source Software: the first one is Spago, the J2EE Framework already released by Engineering Ingegneria Informatica and Sinapsi. Therefore, SpagoBI inherits Spago's features and technical characteristics, using them in its specific context:

- it manages specific Business Intelligence objects as reports, OLAP analysis, Dashboards and Scorecards views and Data Mining models;
- it is focused on data management and on information content and context;

- it supports the BI systems's administrators in the control, validation, certification and distribution process of the Business Intelligence objects.

3 SpagoBI features

According to its development road-map, the main SpagoBI features will be the following:

- **BI Portlets:** every BI object will be distributed for the end-user through portlet technology (JSR 168). In this way portlets and the BI objects will be managed and encapsulated into the portal already chosen for the specific enterprise solution (even if the portal is realized by means of commercial products).
- **Reporting, OLAP Analysis, Data Mining, Dashboards and Scorecards** are the BI objects managed by SpagoBI, everyone with its own execution engine and development environment. SpagoBI manages the production and validation cycles, the parametric activation, the navigation, the results versioning and storage in a similar way although every BI object maintains its distinctive characteristics.
 - **Reports** realize the structured information views; they have a greater diffusion level according to a static structure (.pdf, .xls, .csv, .html, etc). SpagoBI enables the navigation capability between different reports, inheriting the parameters.
 - The multidimensional structures for the **OLAP analysis** add a higher degree of freedom and variability. The analysis axis and the observation measures are structured, this enables to obtain a data examination at various detail levels and from various perspectives, by means of drill-down, drill-across, slice and dice operations.
 - **Data Mining** algorithms and processes (Neural Networks, Decision Trees, etc.) will enable data analysis, with the aim to find out hidden information. SpagoBI support Data Mining models implementations and their results analysis trough the other Business Intelligence objects.
 - In the **Management Performance** context, SpagoBI provides many widgets for dashboards structuring and the parametric evaluation of performance scores.
- **Query-by-example:** it offers a visual mode for data inquiring. It is possible to save the structure as a template for subsequent reports development, or to export the results for external elaborations (ex. cvs, XML).
- **Interaction with source systems:** it provides connectors, protocols and services for bidirectional data exchange with source systems.
- **Meta Repository:** SpagoBI is a really integrated platform instead of a confused set of products thanks to metadata. The meta-repository contains all information about data (technical and business metadata), processes and rules for the platform management.
- **Users profile:** it is possible to differentiate the functionalities access rights according to the user's role.

- **Documents management:** it is a versioned repository for all the relevant results provided by the BI objects, in a scheduled way too. Research and detection functionalities for these documents are provided.
- **Scheduling:** an off-line activation of all transport and/or import/export data processes, document production, storage and destruction, etc. is provided.
- **Workflow:** SpagoBI manages the approval and certification flow for BI objects and for the relative elaborated documents.
- **Administration:** it concerns in a support to the management functionalities of the whole platform.
- **Logging/Auditing:** corresponding to some control services for the platform's functional and performance monitoring.

4 SpagoBI Architecture

SpagoBI inherits the MVC architectural pattern from the J2EE Spago Framework¹ which is used for SpagoBI's development. Therefore, two remarks are relevant for the design of SpagoBI:

- It is possible to decline the MVC architectural pattern on the Business Intelligence aspects;
- metadata play a primary role as focal point for the guide and the control of the entire platform.

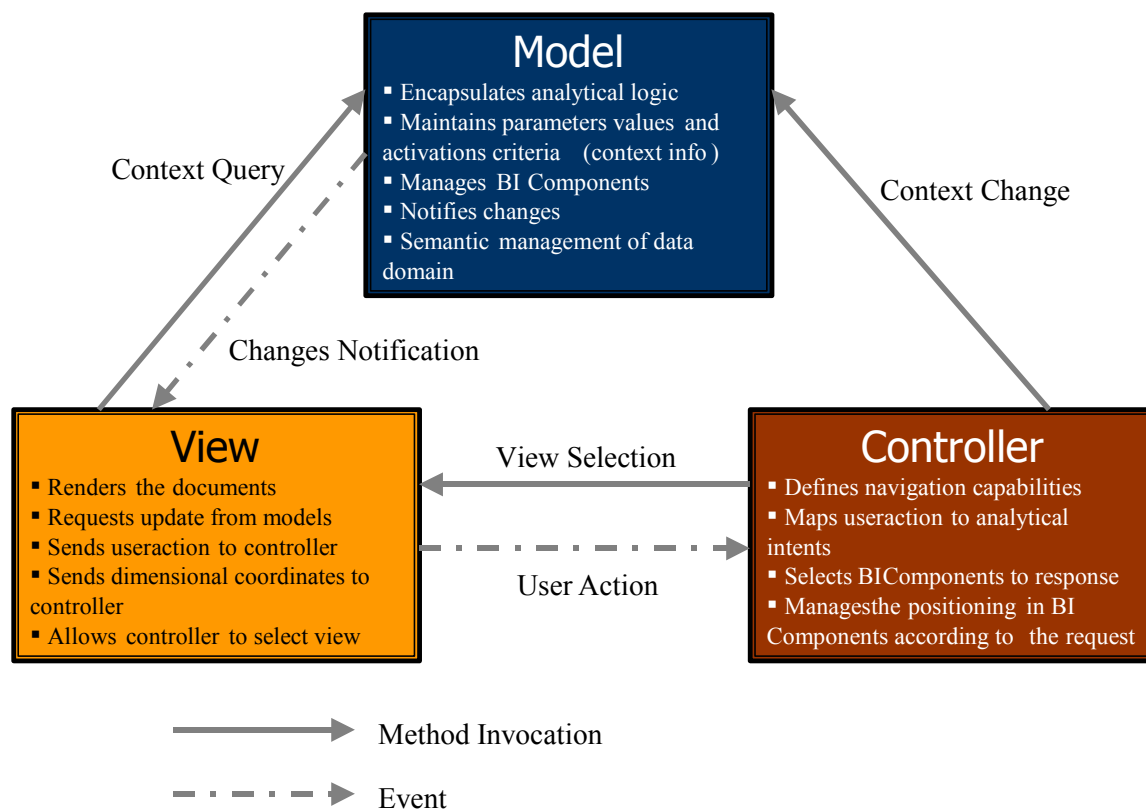
According to these two simple principles, the MVC pattern characterizes itself here in this way:

- **Model:** it's the layer that encapsulates the application logic, which now assumes the means of analytical logic. Business Intelligence specific objects (ex.: Reports, OLAP Analysis, Dashboard, etc.) are handled as generic BI Components with their own control logic and with a specific execution Engine/Server. Anyway, they remain application components which govern the business logic, where business is the analysis.

About the data domain, it locates itself at a Data Warehouse level instead of at an operational level, even if that doesn't avoid the possibility to gain a considerable detail level. Finally, data domain involves metadata level for semantic data management and for the expression of the control logic.

- **View:** adding to the problems of all the web applications concerning data publication and information distribution in this context it is necessary to keep and require documents and data with an historical validity. In analytical logics it is less necessary the manual inputs management by the end user.
- **Controller:** in the Business Intelligence the Controller takes the responsibility for coordination and navigation across specific objects and between different modules. Therefore, it solves also the drill-down, drill-across, slice and dice requests with a less rigid logic than in web applications.

¹ See the Spago documentation for its implementation of MVC pattern.

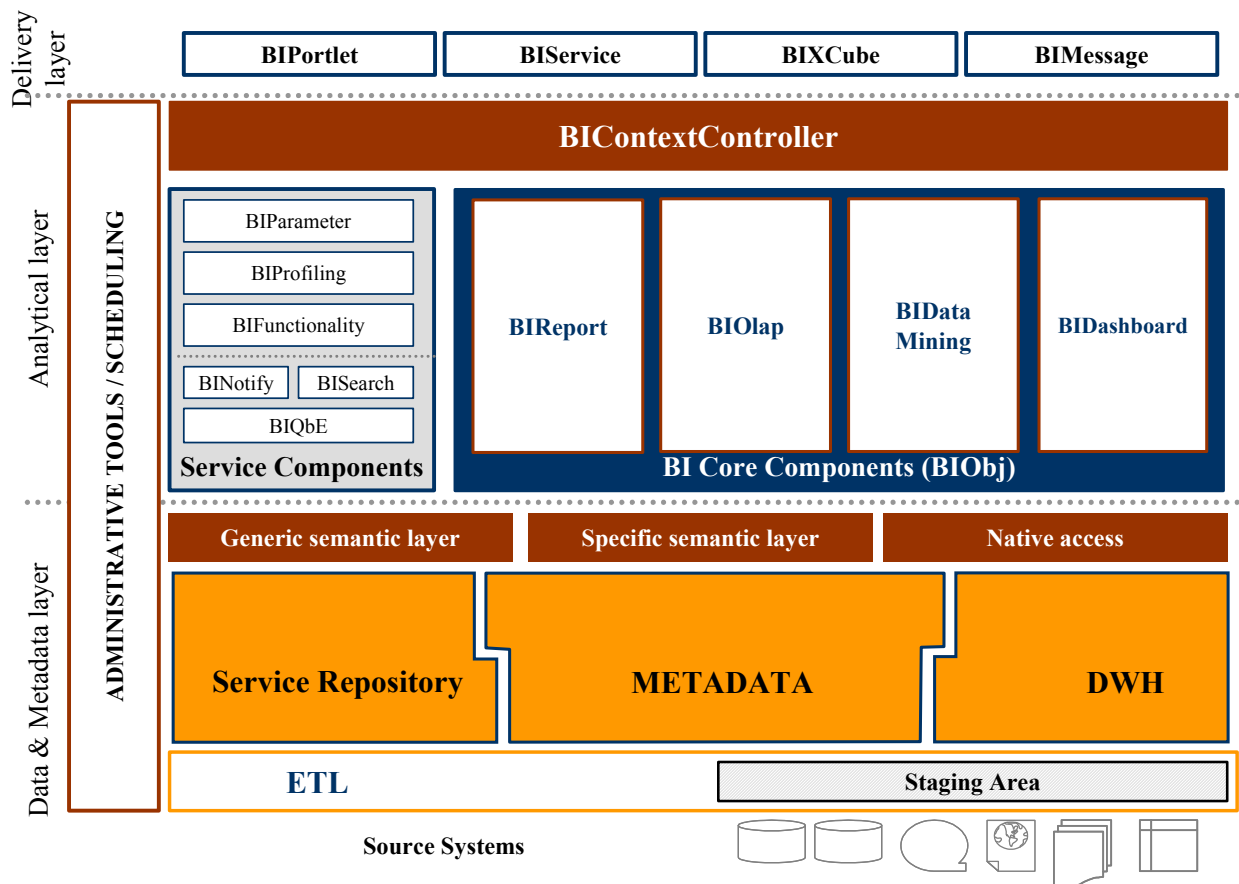


Picture 1 – MVC pattern in BI

We will not continue in the description of how Spago implements the three MVC modules²: we just say that SpagoBI inherits the Spago's settings translating them in its specific way as follows:

- **Delivery layer**, distributing information and analysis models;
- **Analytical layer**, transforming rough data in meaningful information;
- **Data and metadata layer**, receiving data and structuring them for analytical purposes.

² See the Spago documentation for its specific implementation of Model, View and Controller modules.



Picture 2 – SpagoBI architecture

In the **Delivery Layer**, portlets are the first way to access to the BI objects. It's possible to build your custom interface in a new Business Intelligence Portal, or in a part of an existing one, by including the necessary portlets.

You can use Web Services for the interaction with other enterprise applications and with portals too. At last, XML structures and models allow a simple transmission of the information and of the results.

The **Analytical Layer** is the platform core which coordinates all the analytical activities supplying their supporting tools. Its main components are Report, OLAP, Data Mining, Dashboard and Scorecards modules: each of them corresponding to specific functionalities with the same architectural setting; it allows an easy learning and a modular use.

Some service components support the core effectiveness preparing the working environment in its collateral aspects: parameters unified management, filters and domains, query by example capabilities, user and object profiling systems, structuring of categories for documents classification, documents storage and search, approval and management workflow.

The controller takes the responsibility for a successful coordination of the core components interaction and of their interaction with the service components.

The **Data and Metadata Layer** locates itself at data warehouse level and provides a meta-description of its technical aspects and business meanings. A service repository to support document management and object profiling is also provided.

The interaction with the core business components works in a native way or through a semantic interface layer.

Its relation with the source systems is organized by an ETL module, that provides some features for data extraction, transformation and loading.

Finally, **administration** users are supported by many tools for the whole platform configuration and control.

Other information and details about the platform architecture are provided in the specific architectural design document.