

New Trends Towards Process Modelling: Spago4Q

Gabriele Ruffatti, Sergio Oltolina, Daniela Tura
Engineering Ingegneria Informatica
gabriele.ruffatti@eng.it, sergio.oltolina@eng.it, daniela.tura@eng.it

Ernesto Damiani, Carlo Bellettini, Alberto Colombo, Fulvio Frati
University of Milan
damiani@dti.unimi.it, carlo.bellettini@unimi.it,
colombo@dti.unimi.it, frati@dti.unimi.it

ABSTRACT

Spago4Q is a FOSS (Free and Open Source Software) platform, released under GNU LGPL license, for maturity assessment, effectiveness of development software process and quality inspection of the released software: this goal is achieved by evaluating data and measures collected from the project management and development tools with non-invasive techniques. Spago4Q architecture, obtained as a verticalization of SpagoBI (the Business Intelligence Free Platform) is designed in order to be easily adapted to complex organizational contexts, independently from the adopted software development processes, infrastructure tools, measurement and assessment frameworks, supporting companies and organizations both in the certification process and, more in general, in monitoring a formalized development process.

Keywords

Software Process Improvement, Software Quality, Meta-model, CMMI.

INTRODUCTION

In today's software companies, different products are developed following different development processes (waterfall, evolutionary, agile processes, and so forth), making difficult for software assessors to produce uniform evaluations of enterprise-wide process preserving maturity and effectiveness. Traditional process modeling techniques associate specific measurement frameworks to each process; reports generated for different processes are difficult to reconcile in a single company-wide vision. Furthermore, achieving a certification, like CMMI [1,2] or ISO 9001-2000 [3], requires a uniform set of data to be evaluated by assessors or by companies themselves. It is important for managers to create a common environment, which allows to control all aspects of the software development process, producing and giving consistent quality and maturity/capability oriented reports. For these reasons, a new trend in process modeling is emerging, decoupling the development process, the underlying data structure, and the measurement and reporting framework.

Such independence can be reached by adopting a common structure (a *meta-model*), for all aspects of process measurement activity. The definition of a meta-model for the development process leads to the instantiation of individual models for any specific development process, preserving relations consistence between specific instances of the measurement model. In this paper we describe Spago4Q, a new approach to software development process monitoring. Spago4Q integrates an advanced meta-model representation for the process (the Spago4Q meta-model) which makes it fully independent from the specific development process, from the underlying data representation technique, and from certification/measurement frameworks.

As an example, we will outline Spago4Q measurement framework tailored to perform CMMI assessments.

REFERENCE MODEL

In the following sections, the key concepts behind Spago4Q are explained, focusing on details that are interesting in the construction of the underlying meta-model.

MOF Approach

First of all, the definition of the Spago4Q meta-model follows the Meta-Object Facility (MOF) approach proposed by Object Management Group (OMG) [4]. This approach defines the conceptual building blocks to create a meta-model, that generally describes a system, and an instantiation mechanism to produce a model related to any specific situation.

In Spago4Q, three major meta-model components have been identified: *Process*, *Measurement*, and *Assessment* meta-models.

Process Meta-model

The process meta-model has been defined starting from a simplified version of the OMG's SPEM (Software Process Engineering Meta-model) specifications [6], which describes a concrete software development process.

The process meta-model has been designed to be as general as possible, supporting instantiation of both traditional and agile processes. As shown in Fig. 1, it

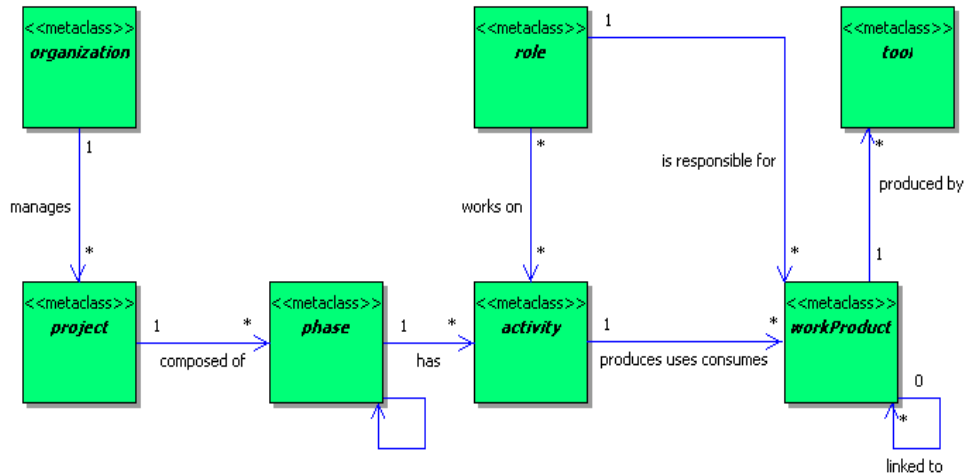


Fig. 1: Meta-model representation of software development process

defines an enterprise development environment as a set of *projects*, with a specific development process composed of *phases*, each of them linked to other *phases* and composed of a set of *activities*. Each *activity* works on a specific set of *workProducts*, over which some specific *roles* act.

In this layer, the meta-model has two main purposes: to define a common structure for any specific development process, and to formalize relations between entities of the measurement and assessment meta-models.

Measurement Meta-model

The measurement meta-model has been defined following the Goal-Question-Metric (GQM) paradigm [7]. Following GQM principles, the meta-model defines three main entities:

- *measurableConcept*, which defines the goals of our analysis, that is the concepts to measure;
- *measurableAttribute* which defines which attributes are to be measured to evaluate a specific goal;
- *KPI (Key Process Indicator)* and *metric*, which defines the operations to apply on measured attributes.

The meta-model also specifies relations between the entities *measurableConcept* and *workProduct*, indicating in which *workProduct* (requirements, source classes, test case, and so forth) the attribute has to be measured. This relation has been formalized at meta-model level, ensuring that it works whatever process model is used.

It is important to remark that any specific measurement model instantiated from this part of the meta-model will define a set of KPIs and metrics specific for the present information needs and of the assessment framework selected by the organization.

Assessment Meta-model

Companies adopt an assessment framework to evaluate the quality of the work execution which, in turn, can be related to the quality of the resulting products.

In Spago4Q, such a framework is formalized in a simple meta-model composed of three entities:

- *category*, container node that gathers goals of assessment in categories;
- *target*, which indicates the target to examine;
- *practice*, which indicates the practice associated to specific metrics.

Spago4Q fully supports CMMI (Capability Maturity Model Integration) [1,2] as assessment framework. Therefore, the keeping of CMMI practices is monitored by specific *KPIs* and *metrics*, examining their values with respect to predefined thresholds.

Spago4Q gives at a glance a reliable snapshot of company state and allows managers and assessors to monitor the adherence of CMMI goals.

SPAGO4Q: ARCHITECTURAL MODEL

The architectural model satisfies the following major requirements:

- open system and compliance to “de facto” standards;
- web-enabled application;
- highly adaptability to various organizational contexts;
- measurement process not bound to the adopted software development process and tools;
- automatic data collection from a set of tools;
- support for a complex system of evaluation;
- secure access and privacy protection management;
- measurement’s knowledge base: set of “library of measurements” and meta-model instances to satisfy the needs of end users, providing a low cost “out of the box” solution.

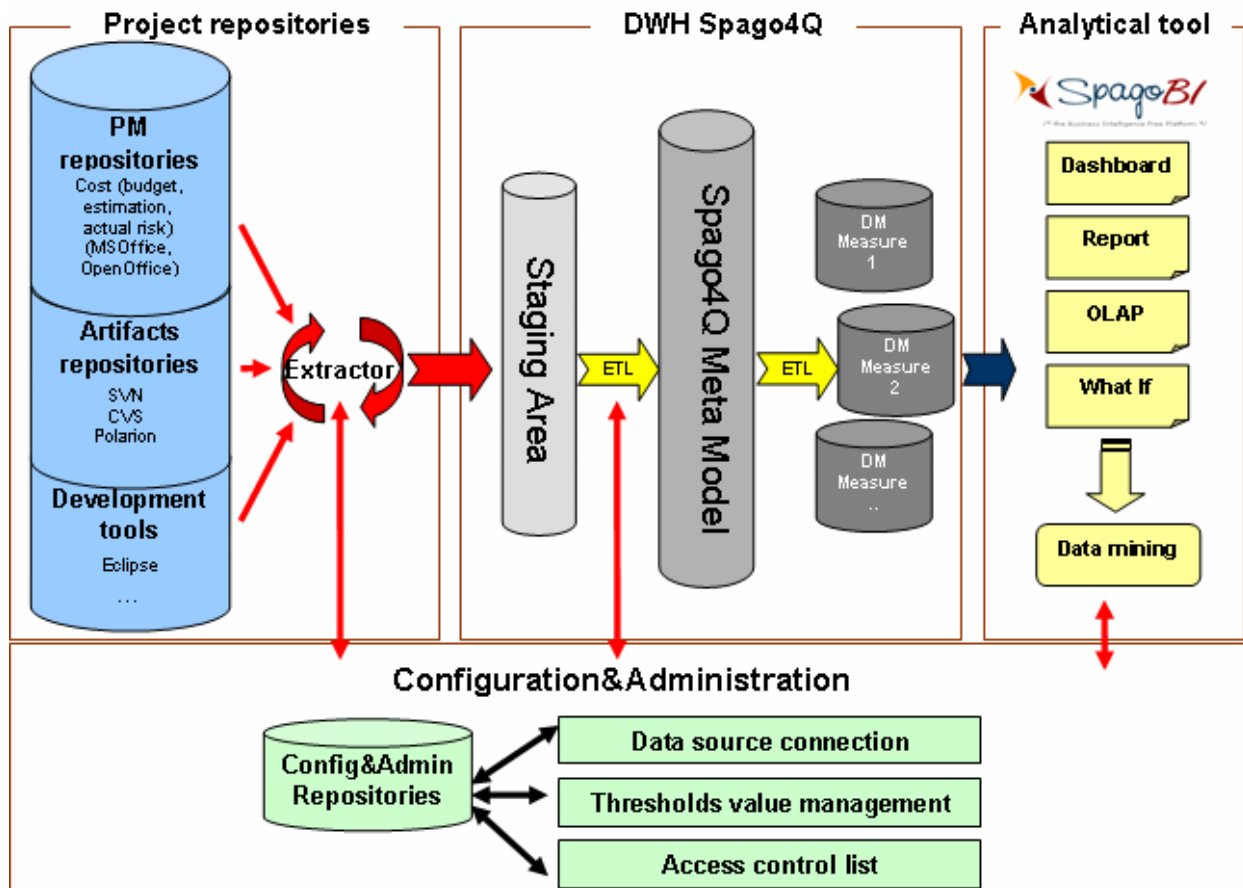


Fig.2 Architectural overview

Figure 2 depicts out Spago4Q architectural overview: the *ETL* (Extract, Transform, Load) procedures extract data from the infrastructure tools and load them into the *DWH-Spago4Q* datawarehouse module, based on the meta-model described in the previous section; *SpagoBI* analyzes data and represents metrics and the *Configuration & Administration* module allows system configuration.

The operational flow starts collecting data needed for chosen KPIs and metrics from the infrastructure tools. The extractor components must be specialized for being properly connected to different repositories and tools adopted in the development process.

ETL's process loads data into *DWH Spago4Q*, the core quality repository, a particular meta-model instance representing the specific development process, measurement and assessment framework.

The datawarehouse [8] structure is designed in order to satisfy the following requirements:

- snowflake schema;
- fact table: one record for every event occurred on a measurable attribute associated to a work-product;
- dimension table: conformed dimensions, shared across every work-products
- historical depth;
- tracking of rejected data.

Spago4Q provides a set of predefined meta-model instances to make easier the measurement process

activity. The instances, ready to use, included in the first Spago4Q beta-release will be:

- specific development processes (UP, SCRUM, waterfall, evolutionary);
- assessment framework (CMMI),
- activity measure areas (requirements, bugs and issues, testing, risks);
- specialized extractors to collect data from a predefined set of tools like: Jira, Bugzilla, Trac, Subversion, CVS, Polarion¹.

In order to complete the assessment process data inserted in the datawarehouse have to be analyzed by Spago4Q analytical component. This module has been implemented as a verticalization of SpagoBI [5], an open source business intelligence platform that covers and satisfies the whole range of BI requirements, both in terms of analysis and data management, administration and security. Using SpagoBI platform to implement the Spago4Q analytical components make easy to represent every KPIs, metrics and the related thresholds as an instance of a particular analytical document type offered by SpagoBI itself (report, OLAP, dashboard, data mining, free enquiry, geo-referenced analysis).

¹ Commercial tool by Emersoft, used in Engineering to achieve CMMI assessment at level 3, now in progress. It also provides an XML format interface for collecting data from generic project management repositories (e.g. risk management repositories)

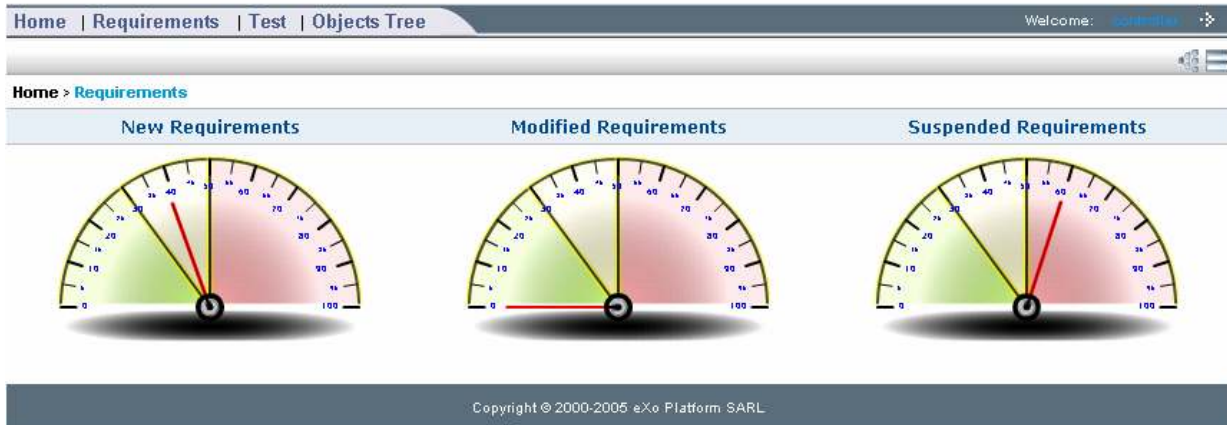


Fig. 3 Dashboard example: Requirements Management

Finally, all the components described above can be properly configured through the *Configuration & Administration module* that provides the following characteristics: *i)* definition of connections to repositories and tools, *ii)* access control list, *iii)* thresholds values management, *iv)* privacy protection management.

The first beta-release of Spago4Q already includes a configurable set of dashboards (Fig.3), reports and OLAP documents to monitor process like: requirements management, bugs and issue tracking, test, risk management, version control.

In conclusion the Spago4Q modular architecture and meta-model design guarantees extensibility towards others infrastructure tools and to further sets of activity measure areas.

CONCLUSIONS AND NEXT STEPS

Spago4Q has been designed to support the activity of assessing the maturity and effectiveness of the software development process and inspecting the quality of the released software.

For the time being, the project is in development phase: the architectural model has been completely defined as well as the meta-model, split into process, measurement and assessment meta-model. A first set of meta-model instances has also been developed (UP, SCRUM, waterfall, evolutionary, CMMI, GQM) and a predefined set of extractor components has been implemented for connecting to different infrastructure tools (Jira, Bugzilla, Trac, Subversion, CVS, Polarion). The project has been accepted by OW2 consortium (www.ow2.org) to be inserted in its forge and in next months it will be possible to both download it (spago4q.ow2.org) and visit the dedicated web site (www.spago4q.org).

A simple prototype is now available to evaluate the capabilities of the platform and, as a sample, the compliance of a specific software project to its user requirements.

Next steps will involve:

- the design of the configuration and administration module;
- the implementation of new extractor components;
- the first beta-release of the Spago4Q (within Dec 15th, 2007).

The collaboration with some European funded research project will be explored. Potential connections (in progress) are been characterized with the following projects: QualiPSO, TEKNE, ETICS.

Spago4Q could be a project really driven by a wide community of both research projects and industrial projects helping them to meet their own goals.

REFERENCES

- [1] B. Chrissis, M. Konrad, and S. Shrum. *CMMI: Guidelines for Process Integration and Product Improvement (2nd Edition)*, SEI Series on Software Engineering, 2006.
- [2] *CMMI Main Page*, www.sei.cmu.edu/cmmi, 2007.
- [3] D. Hoyle *ISO 9000 quality systems Handbook*, Butterworth-Heinemann Ed., US, 2005.
- [4] *MOF Meta-Object Facility - Version 1.4*, www.omg.org/technology/documents/formal/mof.htm, 2007.
- [5] *SpagoBI, the Business Intelligence Free Platform*, www.spagobi.org, 2006.
- [6] *SPEM Software Process Engineering Metamodel*, www.omg.org/technology/documents/formal/spem.htm 2007.
- [7] V.R. Basili. *Software Modeling and Measurement: The Goal Question Metric Paradigm*, Computer Science Technical Report Series, CS-TR-2956 (UMIACSTR-92-96), University of Maryland, College Park, MD, 1992.
- [8] R. Kimball, M. Ross. *The data warehouse toolkit*, Wiley, New York, 2002.