International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

Mapping of Quality Standards

Hind Dahar, Ounsa Roudies

***Abstract*: Quality insurance creates a leverage effect in the global market competition, in particular for software development processes. Thus, companies are brought to use several standards and benchmarks of good practices in management and control of information systems. However, implementing several standards within the same company is difficult and time consuming. In this paper, we propose a Model Driven Engineering approach of building a common repository. It is focused on the common metamodel building. The transformation rules are formalized and illustrated by CMMI model.**

Keywords: About four key words or phrases in alphabetical order, separated by commas.

I. INTRODUCTION

The competition into the global market is based on quality development. This requires companies to use standards and benchmarks of good practices in management and control of information systems. Several standards cover the main activities of IT (information technology) systems such as ISO 9001[1], CMMI [2] and ITIL [3]. Each one has its own strengths but can't satisfy the collection of IT system requirements independently of others [4]. For example, ITIL ensures the post-production quality, while CMMI improves the maturity level of company's processes.

The problem remains in the implementation of several standards within the same company. Several authors believe that this issue is due to heterogeneity of standards that is manifested in their entities and their processes structures [1] [2] [3] [4]. It can also appear in terms of their terminology, size or complexity level [5]. For instance, heterogeneity prevents organizations from interpreting the components of several standards [1] [2] [3] [4]. To solve this problem, they aim to build combinations of best practices, in order to assist quality experts in improving, and optimizing their efforts. However, these combinations can lead to deconstruct models to meet business needs, which makes it difficult to work in multi-model environments [5] [6].

Trying to remedy the inefficiency of the simultaneous use of several standards, we propose a solution focusing on the Mapping of quality standard approach. It is bolstered by Model Driven Engineering (MDE). It aims to align several standards in order to facilitate their integration and to minimize efforts related to their simultaneous usage in the IT field. Furthermore, as appeared in figure 1, the methodology comprises of two phases: Initialization and Use.

The approach's Initialization phase capitalizes on building the Common Repository of quality standard. This model allows to unisfy the fundamental structures of quality standards in order to solve their heterogeneity issue. The Use phase permits to create the meta-model of the target standard, and ensure its transformation to the common repository Model.

Fig. 1. Phases of the mapping approach

To illustrate our model and show its feasibility, we write the rules transformation from CMMI to the common repository by ATL language. Then, we visualize some instances of the CMMI model and the result of these instances in the common repository.

The rest of the paper is organized as follows. The second section describes the building process of a common repository and its theoretical foundation. The third section focuses on the result of the building method. As an experiment, we present in the fourth section the transformation rules from CMMI to our common repository. We finally conclude the paper and share the future works we plan to carry on.

II. BACKGROUND

Our approach focuses on two totally different areas, namely quality standards and the MDE approach. Therefore, before going into the details of our proposal, in this section, we present the necessary background. First, we introduce some quality standards, namely ISO 9001, CMMI and ITIL. Next, we introduce the Model driven engineering approach and describe its main properties used in our approach, namely, meta-models, transformations, and Atlas transformation language.

A. Quality standards

In this overview, we selected three popular quality standards: ISO 9001, CMMI, and ITIL.

ISO 9001 norm comprises a minimum of specific requirements that aim at showing the skills of an organization. These requirements ensure the conformity of the product to the requirements of customers, laws and regulations. The current version considered dates from 2015 [7]. CMMI is created by the Software Engineering Institute (SEI) at Carnegie Mellon University and adopted by the DOD and several American institutions, it has established itself as a standard in the IT field.

Manuscript published on 30 September 2019.
***Correspondence Author(s)**
Hind Dahar*, Univ Mohammed V-Rabat, Emi, Siweb team, EMI, Av Ibn Sina BP 765, Rabat-Agdal,
Ounsa Roudies, Univ Mohammed V-Rabat, Emi, Siweb team, EMI, Av Ibn Sina BP 765, Rabat-Agdal, Maroc,

CMMI identifies three areas of interest, which are CMMI for Development (CMMI-DEV); CMMI for Services (CMMI-SVC) that is dedicated to services management and CMMI for acquisition (CMMI-ACQ). We are interested in CMMI-DEV, that was adopted in 2014. CMMI is a model consisting of a set of best practices that play the role of a guideline for the improvement as well as the assessment of the maturity process within a company [10] [11].

ITIL was created following a call for projects from the UK Ministry of Commerce and established itself as a standard for the delivery of services. ITIL, Information Technology Infrastructure Library, is an efficient methodology in conveying excellent IT. It is a reference of best practices in the quality management of information systems, it possesses an IT infrastructure orientation that focuses on IT production [4]. In this paper, the version 3 of ITIL released in 2013 is considered [12].

B. MDE

MDE (Model Driven Engineering) is a famous approach to deal with the complexity of platforms and the incapability of third generation languages to reduce this complexity and express the domain concepts effectively. MDE technologies combine DSLM (Domain-specific modeling languages) and transformation engines [12].

Meta-models define the relationships among domain concepts and precisely specify the key semantics and constraints associated with these domain concepts.

A model transformation, in model-driven engineering, is a robotized method for altering and creating models[13].

III. PIVOT MM BUILDING METHOD

Our idea is to analyses existing standard structures and concepts in order to build a common repository. It is a type of horizontal mapping of quality standards. It allows transforming the structures of standards into a unified target structure.

We assume that the analysis of leaders quality standards is sufficient to propose a general structure of quality standards, the construction of the common repository went through four stages summarized in Fig2.

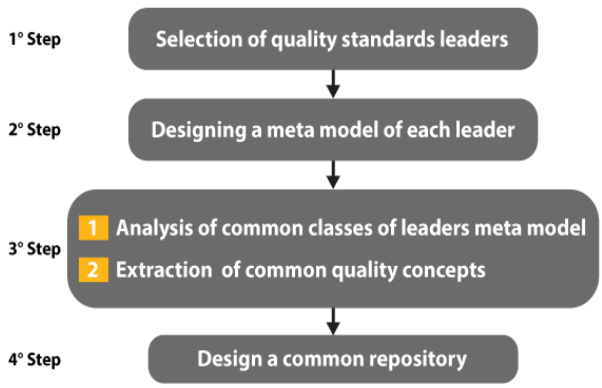

Fig. 2. Method of building pivot model

In the first step, we select the quality standards leaders. The first criterion is popularity and use in IT companies. Among them, we select those that are most representative of their category in order to be able to deduce a solid structure from the quality repository that can manage different types of quality standards.

The second step aims to analyze each leader in order to highlight their main concepts and capture them thus a metamodel.

Finally, the analysis of the three leaders makes it possible to extract the common concept and to establish the mapping between the leaders meta-models. It benefits from the juxtaposition of standards to build a rigid repository. It also identifies concepts that are structurally different but play the same role of quality control regarding the company, i.e. that have shared intentions to solve the problem of heterogeneity generated by their co-deployment. Subsequently, we unify similar concepts and common structures in a common repository. According to MDE, this repository is described by a unified Meta-model that takes into account all the concepts and attributes and keeps all the information related to the standards.

IV. BUILDING THE PIVOT MET

A. Step 1

A literature review led us to identify the following list of the most popular quality standards in use in IT field. Based on the analysis, we select the three most representative standards: ISO 9000, CMMI and ITIL. These three quality standards cover different activity sectors. Unlike ISO 9001, which concerns organizational processes in general, ITIL and CMMI are dedicated to information technology. These standards highlight the different aspects related to challenges faced by existing approaches, namely, heterogeneity, redundancy, etc.

B. Step 2

In this section, we describe the metamodels we propose to model the three standards CMMI, ISO 9001 and ITIL.

ISO 9001: Fig. (3), presents the ISO 9001 meta-model. Its components are organized in *articles*. We distinguish two types of articles, *elementary articles* and *compound articles*. An elementary article contains a list of requirements. Whilst, compound articles are subdivided into several articles. Articles are identified by an identifier *(idArticle)* and a label *(label-Article)*. A requirement characterizes a need to be satisfied in order to obtain ISO 9001 certification. It is identified by an identifier *(idRequirement)*, a label *(labelRequirement)* that contains the requirement text, and a requirement validity *(validityR) which determines whether the requirement is valid or not.*

Retrieval Number: K12990981119/19©BEIESP
DOI: 10.35940/ijitee.K1299.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

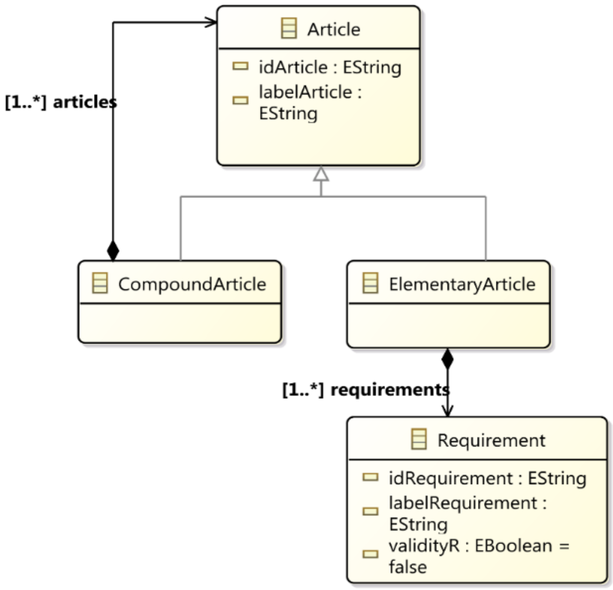

Fig. 3. ISO 9001 metamodel

Fig. (4). illustrates the CMMI meta-model. CMMI is composed of five maturity levels, each one is identified by an identifier (*idM*) that presents the number of the maturity level, and a label (*labelM*) that contains its name. It presents a set of process areas that are identified by an identifier (*idProcess*) containing their name abbreviation, and a label (*labelProcess*) that is the name of the process area. Let us notice that these treatment areas are limited for those specific to the maturity level in question. For example, we will not add the process area corresponding to level 2 to maturity. Process area is associated with goals to be achieved. We distinguish two types of goals. Generic goals belong to all process areas of one level while specific goals are associated to one specific process area. Goals are identified by an identifier (*idGoal*), a label (*labelGoal*) that is the goal text and a goal type (type) that specifies if the goal is specific or generic. The generic and specific goals are associated respectively with generic practices and specific practices that are identified by an identifier (*idPractice*), a label (*labelPractice)* that contains the practice, and a type (*typeP*) that it's depending on the goal type (*type*).

Fig. (5). illustrates the ITIL meta-model. ITIL standard is represented by five books. Each book describes a *Life cycle phase* of the service and is identified by an identifier *(idL)* and a label (*labelL*) that means the phase name. It is built by several *processes*. Those present a succession of *activities*. The *process ITIL* is identified by an identifier (*idP*), a label (*labelP*) that means the process name. *Process activity* is identified by an identifier (*idActivity*), a label (*labelActivity*) that contains the label of the activity and the activity validity as well (*validityA*).

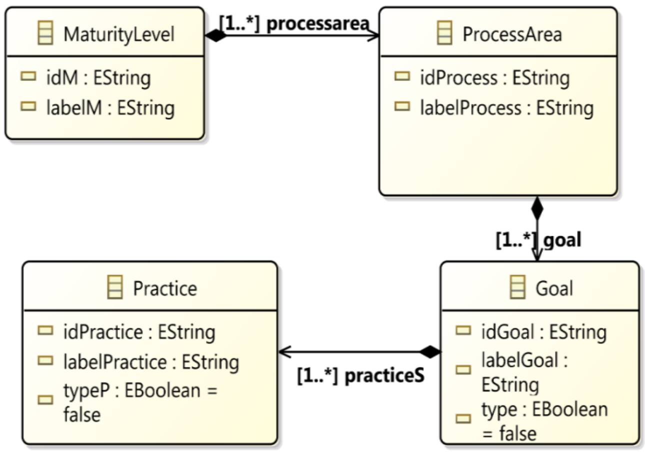

Fig. 4. ISO 9001 metamodel.

C. Step 3

This step is supported by the analysis of the meta model of the three selected leaders. It helps to extract common quality concepts and establish the mapping between them, as shown in Table 1.

Table- I: Mapping of concepts of the three leaders

ISO 9001	*CMMI*	*ITIL*
Article	-	-
Compound article	Maturity level Process area	Life Cycle Phase
Elementary article	Goal	ITIL process
Requirement	Practice	Activity

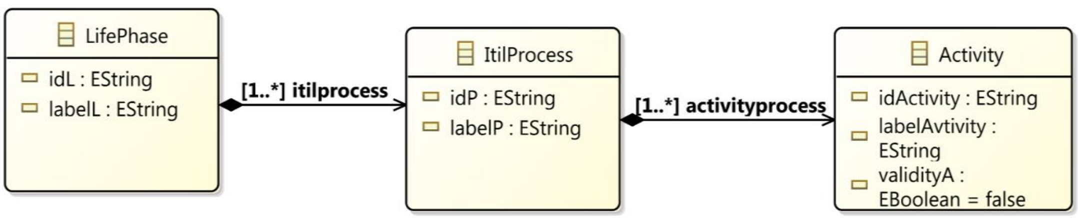

Fig. 5. ITIL metamodel

The table is composed of three columns, each with a leader standard to note ISO 9001, CMMI and ITIL. The lines contain the classes of leaders who present the same concept. For example, at line 3, the concept of requirement of ISO 9001 is similar to the role of the practice in CMMI and activity in ITIL. The article concept of ISO9001 has no correspondence at the level of genericity.

Retrieval Number: K12990981119/19©BEIESP
DOI: 10.35940/ijitee.K1299.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

D. Step 4

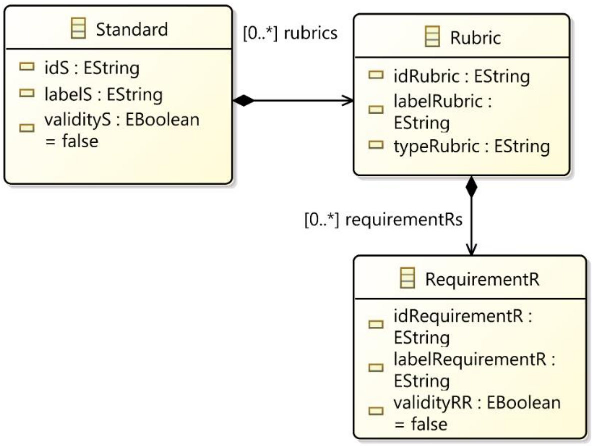

Fig. 6. Common repository metamodel

Fig. (6). presents the Meta model of this repository. The common repository consists of class *Standard* identified by the abbreviation (*idS*) and the name presented as the label of the class (*labelS*), and the standard validity (*validityS*). Each standard is composed by a set of *rubrics*. It enables to subdivide the standard into several domains or specialties. It is defined by an identifier (*idRubric*) and a label (*labelRubric*). The type of the rubric (*typeRubric*) changes from one standard to another: For example "*article*" for ISO 9001, "PA" (process area) for CMMI and life cycle for ITIL. In addition, *Requirements* expresse the need to be satisfied, in order to obtain certification of a standard. It is identified by an identifier (*idRequirementR*), a label (*labelRequirementR*) and a validity (validityRR).

V. TRANSFORMATIONS RULES

A. Transformation process

Fig. (7). presents the transformation process. The mapping between two quality standards is performed thru the mapping of each standard with the common repository. Let us consider a new standard. The preliminary step is to build its meta model. Then, according to MDE, we need to write transformation rules: from standard meta-model to repository meta-model.

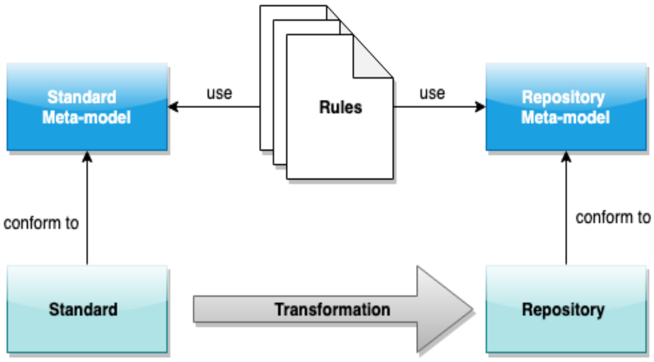

Fig. 7. Transformation process

B. Case study: CMMI

The validation of the common repository is carried out by the transformation from CMMI to a common repository.

Table II presents the transformation rules established from the CMMI to the common repository. The first column of the table presents the source meta-model classes, which is CMMI meta-model. The second column presents the target classes of the common repository meta-model. Each box in this column contains the class of the common repository model that corresponds to the CMMI class of the same line. For example, the *Practice* class is transformed to the *Requirement* class. The third column contains the correspondence between the source classes attributes and target classes attributes, for example, the identifier of the Requirement class (*idRequirementR*) takes the value of the identifier of the Practice class (idPractice), its label (labelRequirementR) takes the value of the practice class label (labelPractice).

Table- II: The transformation rules from CMMI to the common repository

Source	*Target*	*Attribute*
	Standard	id-s := CMMI label-s := Capability maturity model integration validity-s :
Goal	rubric	id-r: id-g label-r : label-g type-r : type-g
Practice	Requirement	id-rp: id-p label-rp : label-p type-rp : type-p

The figure 8 visualizes the transformation rules that must be established between the two meta-models CMMI and Common repository.

.

Retrieval Number: K12990981119/19©BEIESP
DOI: 10.35940/ijitee.K1299.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

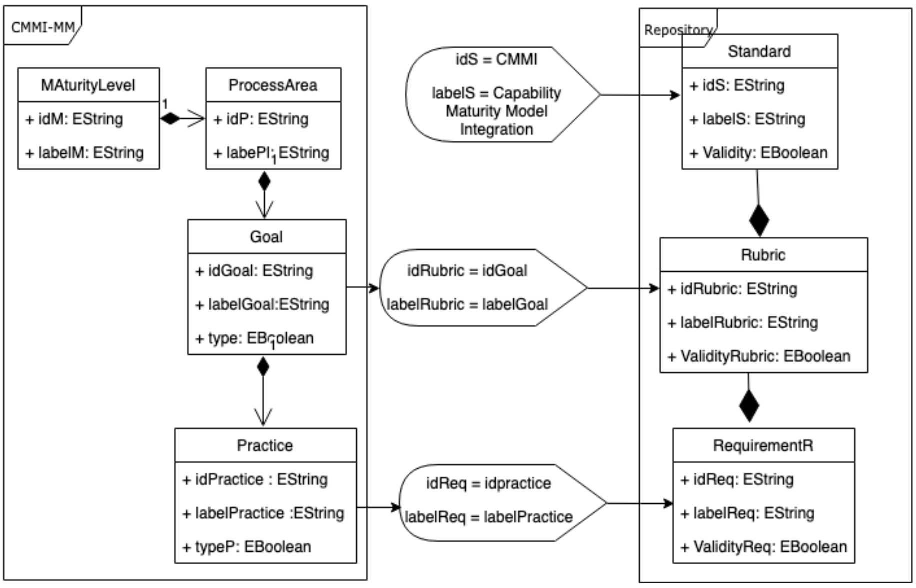

Fig. 8. Transformation model from CMMI to the common repository

C. Experiments

In order to experiment this transformation, we choose an instantiation of the CMMI level 4. It contains two area processes: Organizational Process Performance (OPP) and Quantitative Project Management (QPM).

Table- III: Level 4 of CMMI

Process es area	*Goal*	*Practice*
OPP	SG 1. Establish Performance Baselines and Models.	- SP 1.1 Establish Quality and Process Performance Objectives - SP 1.2 Select Processes - SP 1.3 Establish Process Performance Measures - SP 1.4 Analyze Process Performance and Establish Process Performance Baselines - SP 1.5 Establish Process Performance Models
QPM	SG 1. Prepare for Quantitative Management	- SP 1.1 Establish the Project's Objectives - SP 1.2 Compose the Defined Processes - SP 1.3 Select Subprocesses and Attributes - SP 1.4 Select Measures and Analytic Technique
	SG 2. Quantitatively Manage the Project	- SP 2.1 Monitor the Performance of Selected Subprocesses - SP 2.2 Manage Project Performance - SP 2.3 Perform Root Cause Analysis

We have created a transformation module via ATL. It (ATLAS Transformation Language) is a model change language. It was started in the field of Model-Driven Engineering (MDE), ATL gives approaches to create a set of objective models from a set of source models [9].

Figure 10 presents the transformation modules of the CMMI model to the common repository. The module is named CmmiToRepository, the input of the module presents the CMMI model and the output described the result of the transformation of the CMMI model. The module contains three transformation rules: standards, requirement and rubric. Each of the rules allows to instantiate a class of the output model from the CMMI model. It determines the class source and the corresponding attributes.

```
module CmmiToRepository;
create OUT : MM1 from IN : MM;


rule standard{
 from i2 : MM!ProcessArea
 to
      02 : MM1!Standard (
                idS <- 'CMMI',
                labelS <- 'Capability Mauturity Model Integration' ,
                rubrics <- MM!Goal.allInstancesFrom('IN')
                        )
            }

rule rubric{
 from  I : MM!Goal
 to
       O : MM1!Rubric (
                idRubric <- I.idGoal,
                labelRubric <- I.labelGoal,
                requirementRs <- I.practiceS,
                typeRubric <- I.type
                        )
          }

rule requirementr{
 from  I1 : MM!Practice
 to
       01 : MM1!RequirementR (
                idRequirementR <- I1.idPractice,
                labelRequirementR <- I1.labelPractice,
                validityRR <- I1.validityP
                        )
          }
```

Fig. 9. Transformation rules

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
- <xmi:XMI xmlns:Repository="www.Repository.com" xmlns:xmi="http://www.omg.org/XMI" xmi:version="2.0">
  - <Repository:Standard labelS="Capability Maturity Model Integration" idS="CMMI">
    - <rubrics typeRubric="Specific" labelRubric="Establish performance criteria and models" idRubric="OPP.SG1">
        <requirementRs labelRequirementR="Establish quality and process performance objectives" idRequirementR="OPP.SP 1.1"/>
        <requirementRs labelRequirementR="Select processes" idRequirementR="OPP.SP 1.2"/>
        <requirementRs labelRequirementR="Establish performance measures Process" idRequirementR="OPP.SP 1.3"/>
        <requirementRs labelRequirementR="Analyze Process Performance and establish the basis for the performance process" idRequirementR="OPP.SP 1.4"/>
        <requirementRs labelRequirementR="Establish Performance Models" idRequirementR="OPP.SP 1.5"/>
      </rubrics>
    - <rubrics typeRubric="Specific" labelRubric="Prepare quantitative management " idRubric=" QPM.SG 1">
        <requirementRs labelRequirementR="Establish the objectives of the project" idRequirementR="QPM.SP 1.1"/>
        <requirementRs labelRequirementR=" Composes the defined processes" idRequirementR="QPM.SP 1.2"/>
        <requirementRs labelRequirementR="Select sub-processes and attributes" idRequirementR="QPM.SP 1.3"/>
        <requirementRs labelRequirementR="Select measurements and analytical techniques" idRequirementR="QPM.SP 1.4"/>
      </rubrics>
    - <rubrics typeRubric="Specific" labelRubric="Manage the project Quantitatively" idRubric="QPM.SG 2">
        <requirementRs labelRequirementR="Monitor the performance of selected sub-processes" idRequirementR="QPM.SP 2.1"/>
        <requirementRs labelRequirementR=" Manage project performance" idRequirementR="QPM.SP 2.2"/>
        <requirementRs labelRequirementR=" Effectuer Analyse de la cause" idRequirementR="QPM.SP 2.3"/>
      </rubrics>
    </Repository:Standard>
</xmi:XMI>
```


Fig. 10. repositoryTransformation result

The result is shown in Figure 10 and Table IV, it shows the output of the transformation. The content of the CMMI model is presented under the structure of the common repository.

Table- IV: Transformation result

Standard	*Rubric*	*Requirement*
CMMI	OPP. SG 1. Establish Performance Baselines and Models.	- OPP. SP 1.1 Establish Quality and Process Performance Objectives - OPP. SP 1.2 Select Processes - OPP. SP 1.3 Establish Process Performance Measures - OPP. SP 1.4 Analyze Process Performance and Establish Process Performance Baselines - OPP. SP 1.5 Establish Process Performance Models
	QPM. SG 1. Prepare for Quantitative Management	- QPM. SP 1.1 Establish the Project's Objectives - QPM. SP 1.2 Compose the Defined Processes - QPM. SP 1.3 Select Subprocesses and Attributes - QPM. SP 1.4 Select Measures and Analytic Technique
	QPM. SG 2. Quantitatively Manage the Project	- QPM. SP 2.1 Monitor the Performance of Selected Subprocesses - QPM. SP 2.2 Manage Project Performance - QPM. SP 2.3 Perform Root Cause Analysis

VI. CONCLUSION

The research discussed in this paper, related to solving the problem of heterogeneity and inconsistency of quality standards, is based on an MDE approach. A common repository meta-model is built based on the mapping of a set of popular standards. It plays the role of mediator between two quality standards. This allows the agile alignment of multiple Quality standards. Introducing a new standard implies building its meta-model, and writing the transformation between model and the common repository one. The method offers flexibility and ease of use to analysts and modellers to adapt and contextualize quality standard models based on each other's data. In addition, the use of several standards for the modeling of the common repository model allows to remain compliant with the majority of the best practices used in information systems engineering.

REFERENCES

1. M. Muoz, J. Mejia, and G. P. Gasca-Hurtado, "A Methodology for Establishing Multi-Model Environments in Order to Improve Organizational Software Processes," International Journal of Software Engineering and Knowledge Engineering, vol. 24, no. 06, pp. 909–933, Aug. 2014.
2. N. Kerzazi, "Conceptual alignment between SPEM-based processes and CMMI," in 2015 10th International Conference on Intelligent Systems: Theories and Applications (SITA). Rabat: IEEE, Oct. 2015, pp. 1–9. [Online]. Available: http://ieeexplore.ieee.org/document/7358391/
3. C. Pardo, F. J. Pino, F. Garcia, M. T. Baldassarre, and M. Piattini, "From chaos to the systematic harmonization of multiple reference models: A harmonization framework applied in two case studies," Journal of Systems and Software, vol. 86, no. 1, pp. 125–143, Jan. 2013.
4. C. Pardo, F. J. Pino, F. Garc`ıa, M. Piattini, and M. T. Baldassarre, "An ontology for the harmonization of multiple standards and models," Computer Standards & Interfaces, vol. 34, no. 1, pp. 48–59, Jan. 2012.
5. F. J. Pino-Correa, Universidad del Cauca, M. T. Baldassarre, and Universidad del Cauca, "A 360-degree process improvement approach based on multiple models," Revista Facultad de Ingeniera Universidad de Antioquia, no. 77, Dec. 2015.
6. Dahar H., Roudies O. (2019) Measurement of Co-deployment of IT Quality Standard: Application to ISO9001, CMMI and ITIL. In: Rocha Á., Serrhini M. (eds) Information Systems and Technologies to Support Learning. EMENA-ISTL 2018. Smart Innovation, Systems and Technologies, vol 111. Springer, Cham.
7. Iñaki Heras-Saizarbitoria, Olivier Boiral, Faking ISO 9001 in China: An exploratory study, Business Horizons, Volume 62, Issue 1, 2019, Pages 55-64,ISSN 0007-6813,https://doi.org/10.1016/j.bushor.2018.08.008.
8. PRIATAMA, Yanuar; NUGROHO, Adi; SITOKDANA, Melkior N. N.. EVALUASI TATAKELOLA TEKNOLOGI INFORMASI DI PD BPR BANK BAPAS 69 MAGELANG MENGGUNAKAN ITIL V3 DOMAIN SERVICE STRATEGY. Jurnal MNEMONIC, [S.l.], v. 2, n. 1, p. 28 – 34, jan. 2019. ISSN 2614-4808

Retrieval Number: K12990981119/19©BEIESP
DOI: 10.35940/ijitee.K1299.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

9. Alicia García-Holgado, Francisco José García-Peñalvo,Validation of the learning ecosystem metamodel using transformation rules,Future Generation Computer Systems,Volume 91,2019,Pages 300-310,ISSN 0167-739X, https://doi.org/10.1016/j.future.2018.09.011.
10. YOUNOUSSI, Siham et ROUDIES, Ounsa. Capability and maturity model for Reuse: A comparative study. In : 2016 2nd International Conference on Cloud Computing Technologies and Applications (CloudTech). IEEE, 2016. p. 302-308.
11. YOUNOUSSI, Siham et ROUDIES, Ounsa. A New Reuse Capability and Maturity Model: An Overview. In : Proceedings of the 2018 International Conference on Software Engineering and Information Management. ACM, 2018. p. 26-31.
12. Di Rocco, J., Di Ruscio, D., Härtel, J. et al. Softw Syst Model (2019). https://doi.org/10.1007/s10270-019-00748-
13. Ciccozzi, F., Carlson, J., Pelliccione, P. et al. Softw Syst Model (2019) 18: 7. https://doi.org/10.1007/s10270-017-0589-6

AUTHORS PROFILE

Hind Dahar graduated as an engineer in information technology from the National Institute of Statistics and Applied Economics in Rabat, Morocco. She has worked as a quality engineer in several multinational organizations. She is currently preparing her PhD in software quality at the Mohammedia School of Engineering in Rabat, Morocco. Her area of interest is alignment of quality standards.

Ounsa Roudies is full Professor since 1990 at the Computer science Department of Mohammadia School of Engineering (EMI), Mohammed V University in Rabat, Morocco. Prof. Roudies is head of Siweb research team (Information System and Web). In particular, Prof. Roudies is interested by Reuse and requirements engineering issues. She is chief-editor of e-TI, the electronic Journal of Information Technology (www.revue-eti.net). Besides, she was head of Computer Engineering Department at EMI, and then she was in charge of Software engineering degree and doctoral degree (DESA and UFR DSA-TAL LN).Sen\'es Garc\'ia received her PhD in Accounting at the University of Granada. She is a Lecturer at the Department of Accounting and Finance, University of Granada. Her research interests are related to cultural, institutional and historic accounting and in environmental accounting. She has published research papers at national and international journals, conference proceedings as well as chapters of books.

Retrieval Number: K12990981119/19©BEIESP
DOI: 10.35940/ijitee.K1299.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)