

# QUALITY MODEL TO THE ADAPTIVE GUIDANCE

Hamid Khemissa<sup>1</sup> and Mourad Oussala<sup>2</sup>

<sup>1</sup>Computer Systems Laboratory, Faculty of Electronics and Informatics,  
Computer Science Institute,  
USTHB: University of Science and Technology Houari Boumediene, Algiers;  
Algeria.

<sup>2</sup>Laboratoire des Sciences du Numérique de Nantes (LS2N), Faculty of  
sciences, Nantes University, France.

## **ABSTRACT**

*The need for adaptive guidance systems is now recognized for all software development processes. The new needs generated by the mobility context for software development led these guidance systems to both quality and ability adaptation to the possible variations of the development context. This paper deals with the adaptive guidance quality to satisfy the developer's guidance needs. We propose a quality model to the adaptive guidance. This model offers a more detailed description of the quality factors of guidance service adaptation. This description aims to assess the quality level of each guidance adaptation factor and therefore the evaluation of the adaptive quality guidance services.*

## **KEYWORDS**

*Quality model, Guidance System Quality, Adaptive Guidance, Plasticity.*

## **1. INTRODUCTION**

Due to technological progress, the developer is considered nowadays as a mobile actor operating in various development context using variable platforms. This trend seems interesting, however, it only poses a problem in the ability and quality adaptation to the possible variations of the development context (Garcia and Pacheco, 2009; Kirk *et al.*, 2009).

For this, it is necessary to assist developers and ensure the plasticity of the adaptive guidance systems (Calvary *et al.*, 2002; Coutaz, 2010; Khemissa *et al.*, 2012; Khemissa *et al.*, 2014) with their ability to adapt to the current development context, defined by the triplet (material platform, developer profile, activity context), in respect of their usefulness. Usefulness refers rigorously to quality services offered to developers. It refers to the ability of a guidance system that allows the developer to reach his objective preserving consistency and product quality in software development.

Finally, a quality guidance system is a system capable to satisfy the developer's guidance needs. Therefore, the system quality is estimated as a set of protocols and principles to be applied during the use of the guidance system to meet those needs.

In a first stage, our work is rather focused on the study and synthesis of the limits of the existing

software process modeling environments (Calvary *et al*, 2002; Coutaz, 2010; Khemissa *et al*, 2012). Taking into account specific factors for an adaptive guidance, we have classified these limits through retained factors describing explicitly the basic concepts linked to the adaptive guidance aspect (Khemissa *et al*, 2012; Khemissa *et al*, 2014). To realize the effectiveness of plasticity concept of the guidance system supported by its adaptation ability to current development context, the selected guidance quality factors are defined by:

- ▶ **Guidance core:** The basic guidance is defined as global orientations core regardless the profile of both the activity context and the actor.
- ▶ **Developer profile oriented guidance:** the guidance orientations are defined on the basis that the human actor, regardless his profile, has a central role in the progress of the development process.
- ▶ **Guidance to activity context:** The selection of the appropriate type of guidance is more often not adapted nor suitable to a current activity context.
- ▶ **Guidance types:** the selection of guidance types remains defined in a manual and intuitive way. It depends on the project manager experience and informal personality.
- ▶ **Plasticity of guidance:** the guidance functions are defined and offered on the basis that the human actor always operates on a uniform development context. It should be noted that the plasticity factor is not invoked by the existing software process environments and meta-models. It is typical to our modeling approach of the adaptive guidance.

Based on the specific factors for adaptive guidance, the environments and meta-models considered for a comparative study are: SPEM (OMG, Inc, 2008) and APEL (Estublier *et al*, 2003) considered as the most representative in the software process modeling, RHODES (Coulette *et al*, 2000; Tran *et al*, 2003) that uses basic concepts closest to those introduced by the proposed approach.

According to SPEM, the guidance is a describable element which provides additional information to define the modeling describable elements. However, the proposed guidance is not suitable to the profile components in the development context. The guidance is rather defined in an intuitive way.

ADELE/APEL is designed on reactive database. It proposes a global assistance of proscriptive type without considering the development context profile and automates part of the development process using triggers.

RHODES/PBOOL+ uses an explicit description of a development process. The activities are associated to a guidance system with various scenarios of possible realization.

We noticed well the global guidance aspect and limits for each meta-model. However, the current tendency is that developers would like to have a guidance quality intervention adapted to specific needs according to the characteristics of the current development context.

In this context, we have proposed an approach to define adaptive guidance modeling in software process. It has been described through a meta-model denoted PGM (Plasticity of Guidance Meta model) based on the concepts of development context's profile (Khemissa *et al*, 2012; Khemissa *et al*, 2014). This approach is defined in a Y description of the adaptive guidance. This description will focus on the three considered dimensions defined by the development context, the adaptation form and the provided service.

Each dimension considers several factors to deduce automatically the appropriate guidance service to be provided to developers according to the current context. The description of the first dimension offers an orientation base of the guidance regarding the profile of both the developer and the activity context. The second dimension defines guidance types to consider explicitly in a specific situation of the development context. The third dimension describes the possible adaptation form of guidance core. Finally, the plasticity of guidance is explained by the functional interrelation between these three dimensions. This approach is described schematically as follows:

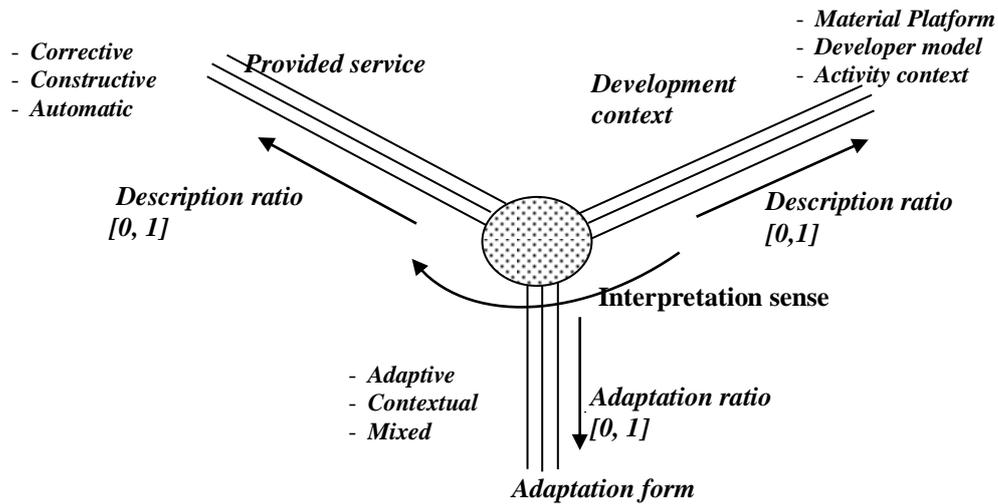


Figure 1. Adaptive guidance in Y description

In this perspective, and with a continuity spirit, we propose in this paper a quality model to the adaptive guidance. This model offers a refined description of the quality factors of the guidance service plasticity and adaptation. This description aims to assess the quality level of each guidance adaptation factor and therefore the evaluation of the adaptive guidance service quality.

## 2. QUALITY MODEL FOR THE ADAPTIVE GUIDANCE

In general, measuring the quality of a guidance system consists then in determining its appropriateness relatively to the guidance adaptation of the functional point of view. Getting a quality measure provides a clear picture of the guidance system and determines its behavior over time in terms of its adaptability to the development context. To have a complete clear image of the guidance system quality, we should define a quality model (Mordal-Manet *et al*, 2011; Mordal-Manet *et al*, 2013).

The most currently known models are hierarchical models that identify the quality principles, starting with the overall requirements and the most general principles to reach the technical criteria and associated metrics. These quality models offer both an overview of the system quality as well as a detailed view according to the considered point of view. They also allow to go from a detailed view to a global view and vice versa (Mordal-Manet *et al*, 2013).

Inspired by the quality model Mc Call (McCall *et al*, 1976), ISO 9126 model (ISO/IEC. Iso/iec 9126-3, 2003), ISO 25010 model (ISO/IEC. Iso/iec 25010, 2011) and Square norm (ISO/IEC. Iso/iec 25000, 2014; Balmas *et al*, 2010) recognized as international standard norms for assessing software quality. The development of our quality model is defined as a four-level model called: point of view-factors-criteria-metrics. It is identified through three points of view associated to the development context representing a global vision of quality.

Each of these point of views is described through five quality factors representing a quality external view. These factors are characterized by eighteen criteria that represent the quality internal view. These criteria are matched with the metrics that evaluate each criterion.

This metric is made on the basis of a process to evaluate quantitatively and semantically each criterion and therefore each of the quality factors and viewpoint in order to deduce every time the adaptive guidance quality according to the considered viewpoint.

The design pattern of the proposed quality model defined by four hierarchical description levels is schematically represented by the following figure.

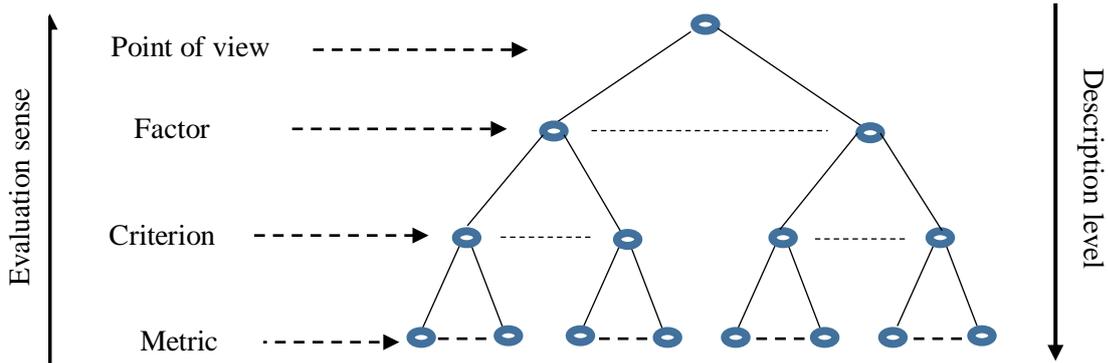


Figure 2. The design pattern of the quality model.

It can be instantiated to describe a specific quality model to a particular domain by describing the set of data related to each level relative to the considered point of view.

In our case, we consider three points of view related to the first dimension considered in our approach ‘‘PGM’’ namely the development context. The quality model for the developer point of view is described by the following diagram.

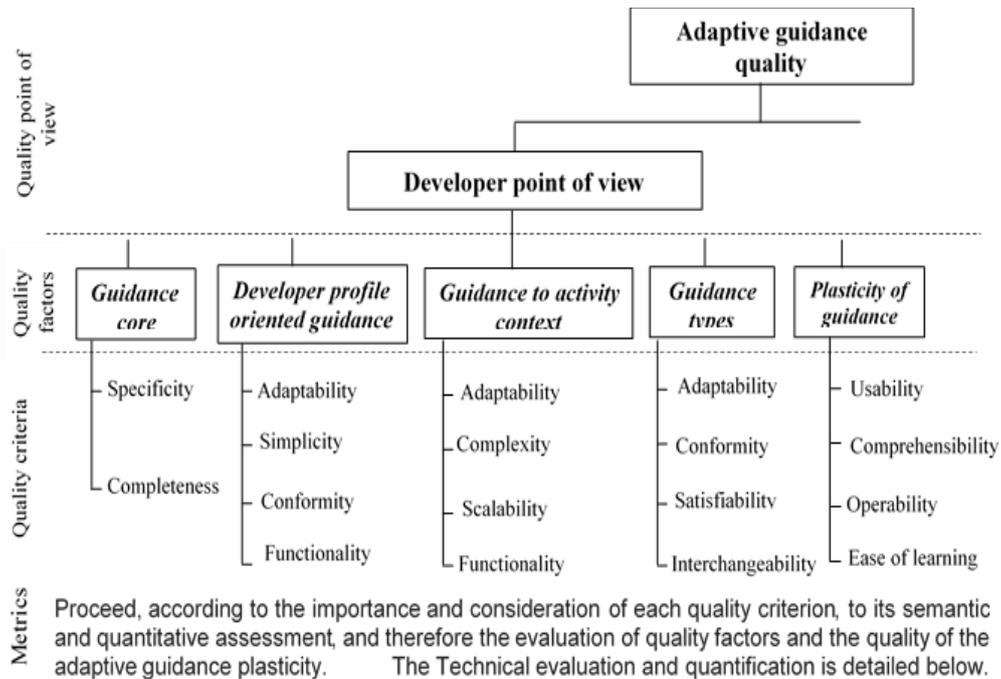


Figure 3. Quality model of the guidance plasticity

The detailed description of our quality model offers a more refined description of our quality factors through the specificity of the corresponding quality criteria. Each of the selected quality factors is described through a set of criteria as follows:

❖ **Guidance core:** this factor is decomposed and evaluated on the basis of the following two criteria:

- **Adaptability:** the degree from which the offered guidance can accommodate with specific situations of activity context.
- **Completeness:** the degree from which the guidance system provides coverage of the whole life cycle of a software process.

❖ **Developer profile oriented guidance:** this factor is discussed relatively to the following four criteria:

- **Adaptability:** the degree from which a guidance system considers, on the basis of the developer profile, all the elements relating to the three dimensions of adaptive guidance.
- **Simplicity:** the degree from which a guidance system can be used to achieve the goals identified by the performer efficiently and satisfiability in a specified activity context.
- **Conformity:** the degree from which a guidance system serves exactly the developer profile needs in a particular activity context.
- **Functionality:** the degree from which a typical system offers guidance services to support the developer needs in specific conditions.

❖ **Guidance to activity context:** this factor is evaluated in relation to the following four criteria:

- **Adaptability:** the degree from which a guidance system considers, based on the current activity context, all elements relating to the three adaptive guidance dimensions.
- **Complexity:** the degree from which a guidance system processes and addresses the needs of the current activity context.
- **Scalability:** the degree from which a guidance system provides the most appropriate behaviour to support the needs of the current context evolution.
- **Functionality:** the degree from which a guidance system offers typical guidance services to address the needs of the current activity context in specific conditions.

❖ **Guidance types :** this factor is appreciated on the basis of the following criteria:

- **Adaptability:** the degree from which a guidance system considers, based on the guidance type, all elements relating to the three adaptive guidance dimensions.
- **Conformity:** the degree from which a guidance system serves exactly the developer needs in a particular activity context.
- **Satisfiability:** the degree from which the offered guidance type ensures the developer needs in the current activity context.

- **Interchangeability:** the degree from which a guidance system supports the consideration of the various guidance services to address the developer needs in the given activity context.

❖ **Plasticity of guidance:** this critical factor targets the degree of a guidance plasticity through the following criteria:

- **Usability:** the degree from which a guidance system can be used on different activity contexts allowing to achieve the goals identified by the performer efficiently and satisfiability.
- **Comprehensibility:** the degree from which a system provides well-structured guidance services to support the developer needs in a given situation.
- **Operability:** the degree from which a system provides a mechanism allowing, at any time, the developer to call the guidance services related to the current activity context.
- **Ease of learning:** the degree from which a guidance system provides a support and learning service to support the concept of adaptive guidance.

### 3. QUALITATIVE EVALUATION PROCESS OF THE ADAPTIVE GUIDANCE

A metric is defined as a quantitative scale and a method which can be employed to determine the value taken by a property or a guidance system criterion.

The evaluation of the adaptive guidance quality is deduced by a practical process at four decomposition levels described by quality metric, quality criterion, quality factor and quality point of view.

This process is defined on the basis of the whole metrics, criteria, quality factors and points of view considered in a given software process environment.

#### 3.1. First level: quality metric

The metric method for the quantitative assessment of each criterion is defined on the basis of the three dimensions considered in the Y description of the adaptive guidance. This metric observes each dimension impact with the involvement or non-consideration of its elements.

This method uses a binary process to note the involvement of each element of the adaptive guidance dimension. The value one " 1 " is associated to each element involved in the criterion evaluation. Since each dimension is defined through three basic components, the expression for evaluating the impact of each dimension is given by:

$$\text{Value Involvement Dimension} = \text{VID} = \Sigma (\text{Value element}_i) / 3 \quad \text{with } i = 1 \text{ to } 3.$$

Finally, the deduction of the quality criterion estimation is based on a mathematical expression combining all elements that define the implication of the three considered dimensions. For the quantitative evaluation, we use the formula of the simple average defined as:

$$\text{Criterion Measure} = (\text{VI context} + \text{VI service} + \text{VI form}) / 3.$$

With:

**VI context:** involvement value of the development context.

**VI service:** involvement value of the guidance service.

**VI form** : involvement value of the adaptation form.

**Application example:** evaluation of the 'Adaptability' criterion for the RHODES environment. The estimation of the 'Adaptability' criterion for the RHODES environment is generated on the basis of the consideration degree of each dimension. Assessing each dimension is made on the basis of its elements involvement. The application of the defined expressions gives us the following result:

$$\mathbf{VI\ context} = \Sigma (0 + 1 + 1) / 3 = 2/3.$$

$$\mathbf{VI\ service} = \Sigma (1 + 0 + 1) / 3 = 2/3.$$

$$\mathbf{VI\ form} = \Sigma (1 + 1 + 0) / 3 = 2/3.$$

$$\mathbf{Metric\ Adaptability} = (\mathbf{VI\ context} + \mathbf{VI\ service} + \mathbf{VI\ form}) / 3. \\ = (2/3 + 2/3 + 2/3) / 3 = \mathbf{0.66}.$$

### 3.2. Second level: quality criterion

Considering the evaluation technique of the proposed model by Boehm (Boehm *et al*, 2009) in the project management, the semantic quantification of each considered criteria is based on its contribution and impact on the adaptive guidance quality.

The semantic quantification process is done through three levels, described by high, medium or low contribution, applying the following rules:

<1/2: high order impact / = 1/2: middle order impact / >1/2: low order impact. Therefore, the numerical estimation of a criterion is done on the data interval [ 0, 1 ].

#### Application Example:

For the purpose of clarity, the following section provides a possible scenario to evaluate semantically the adaptive guidance quality criteria.

The numerical quantification of each criterion is deduced from the application of the defined methods describing the relationship between semantic evaluation and its numerical value.

### 3.3. Third level: quality factor

For each factor, the metric composition is carried at different levels from the measurements obtained at the criterion level.

A simple or weighted average often remains the most used way to compose metrics. The principle of the weighted average aims to promote the most influential criteria. The weight is applied to the criteria according to their influence degree.

In our case, the deduction of the estimated quality factor is based on a mathematical expression by combining the corresponding quality criteria. We use the weighted average formula.

Starting from a developer point of view and for a rigorous influence practice of each quality criterion, we associate the weighting 'Pi' according to the importance of each criterion. The Pi value varies over a range of 1 to n. n represents the largest number of considered criteria to describe a factor quality.

Finally, the computation of the quality factor value 'F<sub>q</sub>' considers both the associated value of criteria quality and the corresponding weighting value for each criterion. This estimate is given by the following formula:

**Adaptive guidance quality** (developer's point of view) =  $\sum F_i * P_{fi} / n$  with  $i = 1$  to  $n$ .

With: **F<sub>i</sub>**: quality factor estimation.

**P<sub>fi</sub>**: associated weighting factor.

**n**: considered factors number.

Finally, the imbrication of the two preceding formulas, allows us to generate a combined expression for estimating the adaptive guidance quality. This combination is formalized by the following expression:

**Adaptive guidance quality** (developer's point of view) =  $\sum (\sum C_i * P_i / n)_j * P_{fj} / m$

With: **i = 1 to n**, **n**: considered criteria number. **j = 1 to m**, **m**: associated factors number.

**C<sub>i</sub>**: quality criterion estimation.

**P<sub>i</sub>**: associated weighting criterion.

**P<sub>fj</sub>**: associated weighting factor.

### 3.4. Adaptive guidance quality

It's also possible to measure the global quality of the adaptive guidance taking into account the three points of view. For this, we also use an average weighting with the necessity of affecting a weighting to the quality of each point of view. Therefore, we generate a layout combining the decomposition levels: criteria, factors and points of view. This combination is deduced by the following expression:

**Global quality of the adaptive guidance** =  $\sum (\sum (\sum C_i * P_i / n)_j * P_{fj} / m) * P_{pv} / 3$

With: **i = 1 to n**, **n**: considered criteria number. **j = 1 to m**, **m**: associated factors number.

**C<sub>i</sub>**: quality criterion estimation.

**P<sub>i</sub>**: associated weighting criterion.

**P<sub>fj</sub>**: associated weighting factor.

**P<sub>pv</sub>**: associated weighting point of view.

## 4. PRACTICAL INTERPRETATION

The practical quality assessment for adaptive guidance is deduced by the quality metric based on the implication of each of the factors associated with the three dimensions considered by our "PGM" approach. This estimation is derived by two phases, the first phase is the semantic evaluation system describing the impact of the quality criteria on the guidance adaptation. The second phase involves the implementation of a digital process based on the formulas and methods defined by our approach through the four quality levels that is: quality metric, quality criteria, quality factor and quality point of view.

This interpretation will address the estimation of the adaptive guidance quality of RHODES environment (Coulette *et al*, 2000; Tran *et al*, 2003). It will focus on semantic evaluation of each criteria defined in relation to its strategy pattern description, its explicit description of the development process and the guidance system of the RHODES environment. The numerical estimation for each criterion is made by applying the defined methods and formulas defining the relationship between the semantic evaluation and numerical value.

Based on our study of the description and functioning of guidance developed by the RHODES environment, the semantic evaluation of each quality criterion is given as follows.

#### 4.1. Quality criteria Evaluation

Based on the involvement principle of each factor associated with the three dimensions considered in our approach, the quantification process is carried by a digital process defined by the following rules:

- Semantic evaluation of "Medium" order is defined by the numerical estimation 1/2.
- Semantic assessments "High" and "Low" order are defined by inversely proportional quantifications, such as:

$$\text{Quantification (semantic evaluation = 1 - Quantification (semantic evaluation of high order))} = \text{Quantification (semantic evaluation of low order)}$$

The fact that the number of involved factors in each dimension is two, giving us an average of  $(3 * (2/3) / 3)$ . The application of this quantification process on the quality criteria of the RHODES environment is defined as follows.

#### 4.2. Quality Factor evaluation

Considering a unique weighting criteria equals to 1, the evaluation of each quality factor on the environment RHODES is as follows:

$$F_q(\text{guidance core}) = (0.66+0.50) / 2 = 0.58.$$

$$F_q(\text{Developer profile oriented guidance}) = (0.66+0.66+0.66+0.50) / 4 = 0.62.$$

$$F_q(\text{guidance to activity context}) = (0.66+0.50+0.66+0.50) / 4 = 0.58.$$

$$F_q(\text{guidance types}) = (0.66+0.50+0.50+0.34) / 4 = 0.50.$$

$$F_q(\text{Plasticity of guidance}) = \text{-----}.$$

Besides, the factor "Plasticity of guidance" is not an invoked factor in RHODES, we note that the RHODES environment covers well the full range of quality factors.

#### 4.3. Quality point of view evaluation

The guidance quality estimation considers both the factors quality value and the corresponding weighting value of each factor. With associated weighting equal to 1, the estimate formula is given as follows:

$$\text{Adaptive guidance quality (developer point of view)} = \sum F_i * P_{fi} / n \text{ with } i = 1 \text{ to } n \\ = (0.58+0.62+0.58+0.50)/4 = 0.58.$$

Finally, we can conclude that the adaptive guidance quality for RHODES environment through only the developer's point of view is estimated at 0.58 and therefore, it's pretty well taken into consideration.

## 5. CONCLUSIONS

Our main purpose in this article is to propose a quality model to the adaptive guidance system for software process modeling. This quality model is highlighted through a detailed description of the quality factors of guidance service adaptation. This description allows to evaluate the

quality level of each guidance adaptation factor in order to deduce the adaptive quality of guidance service.

The evaluation of the adaptive guidance quality is deduced by a practical process at four decomposition levels described by quality metric, quality criterion, quality factor and quality point of view. The developer point of view is described through five quality factors representing a quality external view. These factors are characterized by eighteen criteria representing the quality internal view. These criteria are matched with the metrics that evaluate each criterion. This metric is made on the basis of a process to evaluate quantitatively and semantically each criterion, and therefore each of the quality factors and point of view in order to deduce every time the adaptive guidance quality according to the considered point of view.

A perspective to this work concerns, at first, the necessity to estimate the productivity and cost due to the quality adaptation of guidance system. On another hand, we will also ensure the flexibility and adaptation of the metric system to the possible evolutions of the software process model.

## REFERENCES

- [1] Garcia, I. and Pacheco, C. (2009): Toward Automated Support for Software Process Improvement Initiatives in Small and Medium Size Enterprises. Book chapter. Software Engineering Research, Management and Applications Volume 253, pp. 51–58. C\_Springer-Verlag Berlin Heidelberg. ISBN: 978-3-642-05440-2.
- [2] Kirk, D.C., Macdonell, S.G., and Tempero, E. (2009): Modeling software processes - a focus on objectives, in Proceedings of the Onward. Conference. Orlando FL, USA, ACM Press, pp.941-948.
- [3] Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Souchon, N., Bouillon, L., Florins, M., and Vanderdonckt, J. (2002): Plasticity of User Interfaces: A Revised Reference Framework. In: TAMODIA 2002.
- [4] Coutas, J. (2010): EICS '10. User interface plasticity: model driven engineering to the limit. Proceedings of the 2nd ACM SIGCHI symposium on engineering interactive computing systems. June 2010.
- [5] Khemissa, H., Ahmed-Nacer, M. and Oussalah, M. (2012): Adaptive Guidance based on Context Profile for Software Process Modeling. Information Technology and Computer Science, July 2012, 7, pp 50-60. Volume 4, number 7. DOI: 10.5815/ijitcs.2012.07.07.
- [6] Khemissa, H., Ahmed-Nacer, M. and Oussalah, M. (2014): Plasticity of a Guidance System for Software Process Modeling. First International Conference on Computer Science Information Technology (CoSIT), pp. 49–63, Bangalore, India. CS & IT-CSCP 2014. DOI: 10.5121/csit.2014.4905.
- [7] OMG. Inc. (2008): Software and System Process Engineering Meta-Model Specification version 2.0: Formal/2008-04-01.
- [8] Estublier, J., Villalobos, J., Tuyet lean H, Jamal-Sanlaville, S. AND Vega, G. (2003): An Approach and Framework for Extensible Process Support System. In Proceedings 9th European Workshop on Software Process Technology (EWSPT 2003), Helsinki, Finland, 2003-09-01.
- [9] Coulette B., Crégut X., Dong T.B.T. and Tran D.T., (2000): RHODES, a Process Component Centered Software Engineering Environment”, ICEIS2000, 2nd International Conference on Enterprise Information Systems, Stafford, pp 253-260, July 2000.

- [10] Tran Hanh Nhi, Coulette, B., Crégut, X., Thuy Dong Thi Bich, and Thu Tran Dan. (2003): Modélisation du méta-procédé RHODES avec SPEM. Dans : Recherche Informatique Vietnam-Francophone (RIVF'03), Hanoi, Vietnam.
- [11] Mordal-Manet, K., Laval, J. and Ducasse, S. (2011): Modèles de mesure de la qualité des logiciels, in Evolution et Rénovation des Systèmes Logiciels. Hermès 2011. <Hal 00639279>.
- [12] Mordal-Manet, K., Anquetil, N., Laval, J., Serebrenik, A., Vasilescu, B. and Ducasse, S. (2013) Software quality metrics aggregation in industry. In Journal of Software: Evolution and Process 25 (10) p. 1117—1135, 2013. DOI: 10.1002/smr.1558.
- [13] Mc Call, J., Richards, P. and Walters, G. (1976): Factors in Software Quality. NTIS Springfield.
- [14] ISO/IEC. Iso/iec 9126-3 software engineering -product quality- part 3: Internal metrics, 2003.
- [15] ISO/IEC. Iso/iec 25010-2011 software engineering -product quality- part 1: Quality model, 2011.
- [16] ISO/IEC. Iso/iec 25000-2014 software engineering-software product quality requirement and evaluation, 2014.
- [17] Balmas, F., Bellingard, F., Denier, S., Ducasse, S., Franchet, B., Laval, J., Mordal-Manet, K., and Vaillergues, P. (2010): The Squale Quality Model. INRIA-00533654, Version 1-Second Edition, 8 Nov 2010. <http://www.squale.org/quality-models-site/deliverables.html>
- [18] Boehm, B.W., Abts, C., Brown, A.W., Chulani, S., Clark, B.K., Horowitz, E., Madachy, R., Reifer, D.J., and Bert Steece, B. M. (2009): Software Cost Estimation with COCOMO II. Prentice Hall Edition, ISBN: 0137025769, 978013702576.

## AUTHORS

**Hamid Khemissa** is a full associate professor at Computer Systems Department, Faculty of Electronics and Computer Science, USTHB University, Algiers. He is member of the software engineering team at computer system laboratory LSI, USTHB. His current research interests include Software Process Modeling and Software Modeling Assistance.

**Mourad Chabane Oussalah** is a full Professor of Computer Science at the University of Nantes and the chief of the software architecture modeling Team. His research concerns software architecture, object architecture and their evolution. He worked on several European projects (Esprit, Ist, ...). He is (and was) the leader of national project (France Telecom, Bouygues telecom, Aker-Yard-STX, ...). He earned a BS degree in Mathematics in 1983, and Habilitation thesis from the University of Montpellier in 1992.