Comparison of Software Process Models. A Systematic Literature Review

Christian Cano
Escuela de Graduados
Pontificia Universidad Católica del Perú
Lima, Perú
christian.cano@pucp.edu.pe

Abraham Dávila
Departamento de Ingeniería
Pontificia Universidad Católica del Perú
Lima, Perú
abraham.davila@pucp.edu.pe

Andrés Melgar
Departamento de Ingeniería
Pontificia Universidad Católica del Perú
Lima, Perú
amelgar@pucp.edu.pe

Marcelo Pessoa
Polytechnic School
University of Sao Paulo
Sao Paulo, Brazil
mpessoa@usp.br

***Abstract*— Nowadays, there are several software process models, which fulfill different purposes, approaches and requirements. However, this proliferation causes some confusion in the industry about the benefits or advantages of each proposal. In this context, studies have been conducted to determine the existing equivalence or the extent of coverage between these models having used different approaches to the comparisons. This work aims to present a study of techniques and experiences on comparison of software process models. For this study, a systematic literature review was conducted in relevant databases and available documents finding that there are few works or experiences in this area and it represents an aspect in software engineering the requires a higher level of research and development. Five different methods to compare process models were found and it was identified that the CCT – Comparison Composition Tree method is the unique that have a graphic representation.**

Keywords*— *Software Process Model, Comparison, Systematic Literature Review.

I. INTRODUCTION

The software process models have made a great progress over the last decade, from own proposal (in each organization) to international efforts such as ISO/IEC 12207 [1] or other standards that have achieved international influence as CMMI [2], MoProSoft [3], MPS.Br [4] or the latest ISO/IEC 29110 [5], among others. The newest existing opportunities to achieve a public recognition (certification), the market pushing the companies to develop quality software and the geographic scope of the models set a particular situation for companies like happens in some countries in Latin America to decide which model to use as the basis for their business development plans. A clear example of this situation has been set with the RELAIS Project that sought to introduce two models (MoProSoft and MPS.Br) oriented to small companies that develop software in Mexico, Brazil, Peru and Colombia [6]. Also you should consider that in these countries: (i) there is the influence of ISO 9001 and CMMI; (ii) that have participated in the COMPETISOFT project [7]; and (iii) participate in the development of the new standard ISO/IEC 29110 [8]; so it has four (4) or more software process models in there industries.

On the other hand, even the review made by [9], have found more than 315 standards, guides and other prescriptive documents that are maintained by 46 different organizations. Other authors [10] [11] mention that there is a considerable number of process models. Also, everyone is focused on improving the quality of software but with different nuances depending on the organizations that developed, the application that domain that is oriented [10] [12] [13].

The comparison of the models is an activity in software engineering that has been done in some cases using expert judgment [6] and in other cases using some other techniques that has allowed some level of decomposition process elements and recently using a technique as Composition Trees to help graphically to the analysis of two models [14]. These comparisons have been developed in models like RUP and PMBOK [15], RUP and MoProSoft [16] or ISO 9001 and CMM [17] to find out how much one model covers the other model (coverage), to define an action plan to migrate from one model to another, to develop guidelines for the adoption of a model after being adopted from another model; or even to assess the technical evolution of a model.

In this article, a systematic review is presented, as the focus on the existing comparison methods and the coverage determination of the software process models, and the experience comparison of software process models. Section 2 presents the definition of the systematic review, Section 3, the results of the research; and Section 4, conclusions and future works.

II. SYSTEMATIC LITEATURE REVIEW

This study, follows a formal scheme established in Kitchenham [18] and recommendations about how to propose questions in Santos [19], also has been established a search strategy and analysis. In this section is been presented the following steps:

978-1-4673-9143-6/15/$31.00

A. Overview

This study has used the recommendations of Kitchenham [18] on how to define and carry out a systematic review. For that, it was defined as a study that identify methods and techniques for comparison and coverage determination of software process models. Also, review each one of the identified models, highlighting strengths and weaknesses. The questions that were based on the developed of the guidelines presented by Santos [19], were based on the research that includes the following elements:

- Population: This is the set of items that will be revised. Documents that present and implement methods or technical approaches to the comparison and coverage determination between software process models.
- Intervention: This is what will be evaluated in the set of elements of the population under test. These are methods and technical approaches for comparison and determination of coverage.
- Comparison: elements that serve as a basis for comparison, taking into account the objectives of this work.
- Outcomes: This is the output information that is expected from the research. Comparative study of methods and technical approaches for comparison and determination of coverage between software process models and identification of characteristics of the methods and technical approaches found.

Based on these premises, the following research questions are post:

- What methodologies, methods or techniques (or technical approaches) are used to compare and determine coverage between software process models and the features they have?
- What comparison of software process models have been made in recent years?

The expected end results of the systematic review are the methods or technical approaches to compare process models used in recent years to determine the similarities and differences between two software process models analyzing the distinct aspects of each one.

B. Description and search strategies

The literature research was conducted in two stages. In the first stage the research was performed in electronic databases using the key words that guide the research. The research strings (see Table I) were generated from the combination of key terms and synonyms using OR and AND. These studies were obtained from the following databases:

- IEEE Xplore (http://ieeexplore.ieee.org/)
- Scopus (http://www.scopus.com/)
- ACM Digital Library (http://portal.acm.org)
- ScienceDirect (www.sciencedirect.com)

In the second stage, it was conducted a specific search at the international event SPICE (Conference on Process Improvement and Capability dEtermination in Software, Systems Engineering and Service Management) which related works are related to ISO / IEC 15504 and in the Brazilian Symposiums SBQS (Brazilian Symposium on Software Quality) and SBES (Brazilian Symposium on Software Engineering) where their published works are related to Software Engineering including software process models in order to find relevant articles related to this work. In these conferences from 2007 and 2013 research articles were published. These events record a good number of jobs, which has increased significantly, the presence of foreign researchers. This is reflected in recent articles that are indexed in the area of digital libraries [20]. In the specific case of SBQS, the search is written in English, Portuguese and Spanish items so that the greatest numbers of relevant articles are covering to answer the questions.

TABLE I. KEYWORD USED IN THE STUDY

Field	Value
Population	"ISO/IEC 12207" OR "ISO/IEC 29110" OR "Software Factory" OR "MoProSoft" OR "MPS.BR" OR "CMMI" OR "Software Process Model" OR "Software Company" OR "Software Engineering" OR "Software Process Improvement" OR "Taxonomies" OR "Taxonomy''
Intervention	"Comparison of Software Process Model" OR "Compare Software Process" OR "Software Process Models Comparison" OR "Coverage determination"
Outcomes	"Methodology" OR "Technique" OR "Method"
Search strategy	Population AND Intervention AND Outcomes

TABLE II. SYMPOSIUMS AND EVENTS

Type	Source	Acronyms
Symposiums	Brazilian Symposium of Quality Software	SBQS
Symposiums	Brazilian Symposium of Software Engineering	SBES
Events	International SPICE Conference on Process Improvement and Capability determination in Software, Systems Engineering and Management Services	SPICE

C. Selection criteria for the studies

An initial search of articles returns a large number of studies that are not relevant [18]. Therefore, an iterative and incremental revision is proposed to carry out the systematic review. The iterative term indicates the repetition of one or more activities; the incremental term indicates that the application comes from an initial subset of sources, until it goes to the whole review. This is iterative, since execution (search, information extraction, and visualization of the results) of the systematic review runs entirely on one source of the research, and then the others. Also, is gradual in the sense that the paper (the product) of the systematic review grows and changes with each iteration until it becomes final. Therefore, the inclusion and exclusion criteria should be based on research being related to the subject. Therefore, totally irrelevant studies are discarded at the beginning.

In Addition, it is important to mention that it has not been considered Agile Methods and software process models like

RUP. The first one because we made a previous search looking for some information about if it would be good consider agile methods into the research and we got a background of information describing agile methods as generic methods. These methods cannot be compared with methods like CMMI, MoProSoft, etc. because they are structurally different, However, in future research, it would be interesting to find a way to compare agile methods against methods like CMMI or ISO/IEC 29110. On the other hand, the second one, related to RUP method, this one has not been considered because it is one of the most common software model process and most of the time it is included in software process model search. This premise is supported by reference [21] where a comparison experience is shown using RUP that we got in the search without including RUP option in the search string.

The inclusion of a document is determined by the relevance in relation to the questions asked by analyzing the title, abstract, keywords, conclusions and in some cases a review of the introduction. As pointed out above, the following criteria were proposed:

- Inclusion criteria 01: Will be taken into consideration account jobs and related literature that have methods and technical approaches to compare and determine the coverage between software process models. Also, articles that contain comparison experiences of software process models will be considered.
- Inclusion criteria 02: For specific topics such as comparing analysis that review articles since 2003.
- Exclusion criteria 01: Priority will be given to items with no more than 5 years old. However, for general topics as process models will be reviewed articles since 1993.

III. Study Results

Result of the literature reviewed is presented here.

A. Classification studies

The search procedure produced (see Table III), after a preliminary purification, 263 studies, of which 121 studies that were not repeated, is not taken into account if the same item was found in another database. From this group 31 were selected for their relevance and 22 were selected as relevant primary studies. From the initial selection of 31 articles, studies were selected on the basis of a close reading of titles, abstracts, keywords, conclusions and future works. One percentage analysis was performed to determine the trend of publications per year (see Table IV) and also an analysis of publications by source type (see Fig. 1).

In Table V, the research methods used in the search group of 22 papers, which were selected as primary studies. Also, Table V shows the distribution by type of article using the 22 primary studies, in which 40.9% researches are comparison, which have been used to analyze the techniques and methods applied; and the other 59.1% remaining, are case, which show concepts about some methods and techniques used at the comparison experiences.

TABLE III. Search Procedure

Database	Studies					Percentage (%)
	Search Date	*Discovered*	*Not repeated*	*Relevant*	*Primary*	
Scopus	11/2014	19	17	5	3	13.6
IEEE Explore	11/2014	24	11	5	3	13.6
ACM	11/2014	131	41	4	2	9.1
Science Direct	11/2014	73	36	6	3	13.6
Conferences	11/2014	16	16	11	9	40.9
	Total	263	121	31	22	100.

TABLE IV. Trend of Publications

Year	Percentage (%)	Frequency
1997	3.23%	1
1999	3.23%	1
2000	3.23%	1
2001	6.45%	2
2004	3.23%	1
2006	3.23%	1
2007	6.45%	2
2008	9.68%	3
2009	16.13%	5
2010	12.90%	4
2011	9.68%	3
2012	19.35%	6
2013	3.23%	1

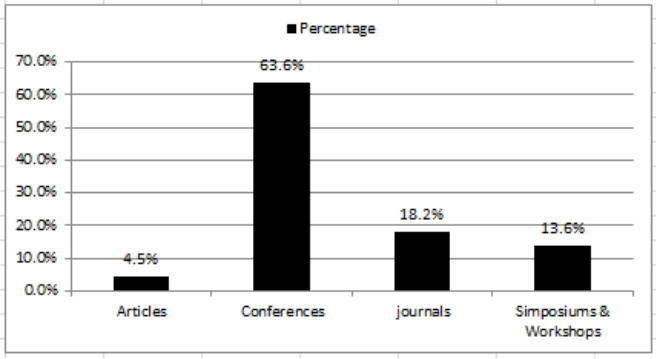

Fig. 1. Publications by source type

TABLE V. Research Methods Founded

Research method	Frequency	Percentage (%)
Comparison experiences	9	40.9%
Case studies	13	59.1%

B. Methodologies or technical approaches identified for comparison process

Based on the evidence to answer the question: What methodologies, methods or techniques (or technical approaches) exist for comparison and determination of coverage between software process models and what features they have? We have identified five methods to make comparison of software process models: (i) "Descriptive Comparison Method" (ii) “Software Analysis Method - SAM” (iii) "Correspondence analysis of process elements", (iv) “Taxonomy” and (v) based on the technique of Composition Tree.

The "Descriptive Comparison Method" used in [22] [23] [24] is based on a systematic observation and detailed description of a software process model. The process followed is as follows. First, it describes in detail the model highlighting aspects such as basic principles, complexity, necessary requirements, cost, and general concepts of the model. Second, it describes in detail the advantages provided by the model, as well as the disadvantages or shortcomings that must be taken into account when deciding to use this model, including a risk analysis. Third, it identify all the aspects or characteristics in common to all models to compare, and then list them including the advantages and disadvantages encountered in each one. Finally, make a table where the rows are the common aspects, advantages and disadvantages. And columns represent each model seeking to determine the relationship between the elements of software process models and the performance of quantification between the elements compared.

The Analysis Method of Software (SAM) used in [25] [26] is introduced to promote better understanding of software processes. This includes their similarities, differences and relations. SAM involves three main steps: (1) Elaboration, which is based in covering all details specified in the base references. A base reference is ideally the standard reference for the process definition. The elaboration usually follows the top-down approach. The final elaboration is determined either when all details in the base references are captured and/or when each element in the elaborated model reach atomic state. An atomic element is the smallest element which cannot be further divided [25]. (2) Normalization is a step that normalizes each term in order to be able to compare on an equal footing. Because, each process element needs to be defined using the same terminology standard. This can be done using a process dictionary. In this stage, SAM, automatically determines and replaces non-standard terms by their corresponding standard terms from the previously defined process dictionary. For example the term “user stories” defined in XP is replaced by “user requirements” [25]. (3) Abstraction, this is the final step where all similar activities are combined to provide an abstract view of each process model. Through the abstraction step, SAM establishes the baseline for process comparison by relating normalized items to their root elements. The normalized terms are further abstracted into root terms. This allows the process engineer to recognize the root of the activities. So, various process models can be compared [26].

The "Correspondence Analysis of Process Elements" used in [15] [17] [21] [16] [27] consists in performing an exploratory data analysis, in order to summarize a large amount of data, with the minimum information as possible. The main objective is to reveal the structure of data on each element of the process, through a decomposition of elements at the lowest level, then analyze and determine the relationship, literally, between process elements compared. To do so, it requires an understanding of the structures of the two models to be compared, considering key factors as objectives, activities, inputs, outputs, purposes, results and other processing elements. These processing elements must be described in detail, and then establish a correspondence between the elements of each process. This correspondence does not always involve one to one relationship between them, or imply that it is complete. In some cases, a processing element may be split into various levels of abstraction for proper correspondence between elements. Finally, the coverage of a process model over another is calculated by the following two rules [16], one of which is the multiplicity between elements (since one or many elements of a model does not necessarily cover fully to an element of another model) and the second rule is the covering (see ref. 5, 7), which consists in assigning a punctuation in function to the level of coverage achieved.

The taxonomy [28] [29] [30] [31], is based on the descriptive comparison method getting a list of characteristics regarded as important for describing SPI Frameworks making a short tabular presentation, which requires each characteristic to have only a short description. More general characteristics would need longer descriptions, which could not easily fit in small table cells. In addition, the overlap between some of the characteristics is important to capture small variations among the frameworks [29]. A characteristic descriptions are been grouped in 5 categories to ease readability and comprehension. (1) General category, this group of characteristics describes general attributes or features of SPI frameworks. Such attributes are often specific to each framework and frequently related to how the framework is constructed or designed. (2) Process category, this group of characteristics describes how the SPI framework is used (3) Organization category, this group of characteristics is related to attributes of the organization and environment in which the SPI framework is used. In others words, it refers to the kind of people who is involved on this and the kind of organization for which is applicable this framework. (4) Quality category, this is related to the quality dimension by pointing out aspects such as how progression is measured, whose quality perspective is employed and what that means in terms of quality indicators and casual relations. (5) Result category, this group of characteristics describes the results of employing an SPI framework, but also the costs of reaching these results and the methods used to validate them [29]. Finally, we have to compare SPI Frameworks among categories of each one.

The comparison using Composition Tree (CT) used in [32] is adapted based on a technique of the requirement engineering to describe graphically the composition of a component of a software system. This method provides graphically summary information, very useful, including states, attributes and relationships on the software system [14]. Therefore, it has an easy way to identify the similarities and differences between related processes. The process followed is as follows: First the main elements of a process model are identified, which are purposes and outcomes. Second, read through the purpose and outcomes, and make a complete and consistent list of nouns and acronyms, which are usually components or attributes of components. Third, starting from the process purpose state, identify the components and their state and draw the initial CT. Finally, read each outcome one by one, to identify the components, states, relationship and attributes and then integrate the information in the CT., having all information is integrated in one graph, so the relationships between different parts become visible. The information of each component is arranged in one place so it will be easier to retrieve and

generally, the graphical version of the process may have less ambiguity, may be easier to understand and easier for people to identify process defects [14].

In Table VI, are shown the main features of each method or technique, which allows you to view the differences between the methods found. Thus can be identified possible advantage of one method over another.

TABLE VI. MAIN CHARACTERISTICS OF THE METHODS FOUNDED

Methods, methodologies or techniques	Necessity of expert judgment	Comparison type	simultaneous comparisons	Systematic methodology
Descriptive comparison method	Yes	Unidirectional	2	No
Analysis Method of Software (SAM)	not necessarily	Unidirectional	2	Yes
Correspondence analysis of process elements	not necessarily	Unidirectional	2	Yes
The Taxonomy	Yes	Bidirectional	2	Yes
Comparison Composition Trees (CCT)	not necessarily	Bidirectional	2 or more	Yes

First, the descriptive comparison method and the taxonomy are characterized by the need for a judgment or expert interpretation about the process models that are being compared. However, SAM, the correspondence analysis method of processing elements and CCT do not need necessarily an expert in the models to be compared, because there is a systematic process established in their methodology.

Second, the descriptive comparison method, SAM and the method of correspondence analysis of processing elements allow an unidirectional comparison, i.e. if you want to compare a model A against a model B, first it must be done by comparing the model A respect to the model B, then a comparison of model B respect to the model A. In contrast, the method of the taxonomy and CCT allows a bidirectional comparison, i.e. if you want to compare the model A against the model B, using only once the algorithm CCT, you will obtain a result of overall comparison without reapplying the method.

Third, the numbers of simultaneous comparisons that can be performed are limited to a total of two for the case of descriptive comparison method, SAM, correspondence analysis of process elements. However, the method of CCT has a great versatility because it permits to compare two or more methods simultaneous graphically.

Finally, each comparison methodology is analyzed to determine a different level of objectivity in the final comparison result, because it depends on the level of interpretation on the analyzed data. On the other hand, it is important to consider whether the methodology used provides a systematic approach to use. In this case, descriptive methodology does not provide a well-defined systematic approach. However, SAM, the methods of analysis of correspondence of process elements, the taxonomy and CCT provide a systematic approach, because they are based on the concept of granularity. In other words, seek to decompose the elements at the lowest possible level.

C. *Cases or comparison experiences of software process models*

In Table VII, are showed the comparison experiences that were founded, where it is used each founded method.

TABLE VII. COMPARISON EXPERIENCES BY TYPE OF METHODS FOUNDED

Methods, methodologies or techniques	Comparison experiences
Descriptive Comparison method	- ISO 9001 vs. CMM [17] - Rational Unified Process (RUP) vs. Microsoft solutions Framework (MSF) [21]
Analysis Method of Software (SAM)	- Waterfall Model vs. Spiral Model [25] [26]
Correspondence analysis of process elements	- Rational Unified Process (RUP) vs. MoProSoft [16] - MPS.BR vs. MoProSoft [27]
The taxonomy	- TQM, CMM v1.1, ISO 9000, ISO/IEC15504(SPICE), IP/EF/GQM, and SPIQ [28] [29] [30] [31]
Comparison Composition Trees (CCT)	- ISO/IEC 29110-5-1 vs ISO/IEC 12207 [32]

In reference [17] it is shown a direct comparison between ISO 9001 and CMM, which shows that software process models with a high rate of adoption in contexts for which they were created not necessarily have a direct correspondence of similarities. It is verified in this work. In reference [21], IBM does the same job but between Rational Unified Process and Microsoft Solutions Framework, seeking to find the similarities and differences, advantages and disadvantages of each one. In references [25] [26], it is shown an interesting method which is used to compare two software process development descriptively with a systematic approach. In references [16] [27], with the proliferation of models for small organizations appears the necessity to compare these new models. In this case, is compared MoProSoft vs. RUP and MPS.BR model, respectively. The similarities are very prominent, and show clear evidence that can be applied to different contexts and markets. By the way, the taxonomy method is really used to compare software process improvement. In this case, is compared CMM against ISO9000. It's an interesting method, but is necessary to have some tools like a data dictionary for process. Finally, the new formal method formal of a graphical notation method, called Composition Trees, has basically one work with one application, which can be analyzed in reference [32]. That paper seeks to validate the input profile of the ISO/IEC 29110 Entry Profile against its counterpart in the ISO/IEC 12207. This validation is done using the method of Composition Trees through a direct comparison of graphical notations, emphasizing clarity and great contribution to not show ambiguity in the outcome of the comparison.

IV. CONCLUSIONS AND FUTURE WORKS

In this article the results of a systematic review of 22 articles, as a primary studies, were analyzed taking into account other relevant articles related to the research topic. The research presents a list of the characteristics of research studies

published by year, type of research, methods and techniques used which is a practice that has been little studied. Furthermore, as noted above, it is necessary to perform work in more depth on this subject so that it can develop a formal scheme and automated work as far as possible, on the basis of CT and coverage rules. As a future work it is planned to develop the support to the measure of the level of coverage of a software process model respect to others.

ACKNOWLEDGMENT

This work is framed within ProCal-ProSer Contract N° 210-FINCYT-IA-2013: "Determinación de factores que influyen en la PROductividad y CALidad en organizaciones que desarrollan PROductos software y ofrecen SERvicios software utilizando como base normas ISO en pequeñas organizaciones." and, the Department of Engineering of Pontificia Universidad Católica del Perú.

REFERENCES

[1] ISO, "ISO/IEC 12207:2008 Systems and software engineering — Software life cycle processes. 2nd. edition," ISO, Geneva, 2008.

[2] SEI, "CMMI for Development, Version 1.3. CMU/SEI-2010-TR-033," Sofwtare Engineering Institute. CMU, Pittsburgh.

[3] H. Oktaba, C. Alquicira, A. Su, A. Martínez, G. Quintanilla, M. Ruvalcaba, F. López Lira, M. Rivera, M. Orozco, Y. Fernández y M. Flores, "Modelo de Procesos para la Industria del Software. Mo-ProSoft. Versión 1.3," UNAM, Ciudad de México, 2005.

[4] K. Weber, A. Rocha, Â. Alves, A. Ayala, A. Gonçalves, B. Paret, C. Salviano, C. Machado, D. Scalet, D. Petit, E. Araújo, M. Girão, K. Oliveira, L. Oliveira, M. Amaral, R. Endriss y T. Maciel, "Modelo de Referência para Melhoria de Processo de Software: uma abordagem brasileira," de *XXX Conferencia Latinoamericana de Informatica (CLEI2004)*, Arequipa, 2004.

[5] ISO, "ISO/IEC 29110-1:2011 Software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 1: Overview," ISO, Geneva, 2011.

[6] RELAIS, "Red Latinoamericana de la Industria de Software," RELAIS, 2010. [En línea]. Available: www.relais.mx. [Último acceso: 14 03 2015].

[7] CYTED, "Proyecto: Mejora de procesos para fomentar la competitividad de la pequeña y mediana industria de software de Iberoamérica (COMPETISOFT)," CYTED, 01 2006. [En línea]. Available: http://www.cyted.org/?q=es/detalle_proyecto&un=291. [Último acceso: 14 03 2015].

[8] ISO, "ISO/IEC TR 29110-5-1-2:2011 Software engineering -- Lifecycle profiles for Very Small Entities (VSEs) -- Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile," ISO, Geneve, 2011.

[9] J. Moore, "Integrated collection of software engineering standards," *IEEE Software,* vol. 16, nº 6, pp. 51-57, 1999.

[10] S. Sheard, "Evolution of the Framework Quagmire," *IEE Computer Magazine,* vol. 34, nº 7, pp. 96-98, 2001.

[11] E. Ericsson, P. Gustafsson, D. Hook, L. Marcks Von y W. Rocha, "Process improvement framework evaluation," *Management Science and Engineering International Conference,* pp. 319-326, 2010.

[12] S. Sheard, "The Frameworks Quagmire," *Crosstalk: The Journal of Defense Software Engineering,* vol. 10, nº 9, 1997.

[13] L. Ferreira, R. Machado y M. Paulk, "Size and Complexity Attributes for Multimodel Improvement Framework Taxonomy," de *36th EUROMICRO*, 2010.

[14] L. Wen, D. Tuffley y T. Rout, "Using Composition Trees to Model and Compare software Process," *Proceedings of 11th International conference. SPICE2011,* pp. 1-15, 2011.

[15] S. Charboneau, "Software Project Management" - A Mapping between RUP and the PMBOK," *IBM Rational Edge,* 2004.

[16] K. Cánepa y A. Dávila, "Mapeo de los procesos de RUP respecto a MoProSoft," *Actas de la JIISIC,* pp. 139-146, 2008.

[17] M. Paulk, "How ISO 9001 compares with the CMM," *IEEE Software,* pp. 74-83, 1995.

[18] B. Kitchenham y S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 9 Jul 2007. [En línea]. Available: http://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf.

[19] C. Santos, C. Pimenta y M. Nobre, "A estratégia PICOC para a construção da pergunta de pesquisa e busca de evidências," *Revista Latino-Americana de Enfermagem,* vol. 15, nº 3, pp. 508-511, 2007.

[20] R. Prikladnicki, S. Marczak, T. Conte, J. De Souza, J. Audy, A. Kroll, A. Marques y A. Orsoletta, "A Evolução e o Impacto da Pesquisa em Desenvolvimento Distribuído de Software no Brasil," de *XXV Simpósio Brasileiro de Engenharia de Software*, São Paulo, 2011.

[21] S. Segi, "Comparing the Rational Unified Process and Microsoft Solutions Framework," IBM Developerworks, 2007.

[22] S. Maheshwari y J. Dinesh, "A Comparative Analysis of Different types of Models in Software Development Life Cycle," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 2, nº 5, 2012.

[23] A. Mujumdar, G. Masiwal y P. Chawan, "Analysis of various Software Process Models," *International Journal of Engineering Research and Applications (IJERA),* vol. 2, nº 3, pp. 2015-2021, 2012.

[24] P. Vohra y A. Singh, "A Contrast and Comparison of Modern Software Process Models," *International Conference on Advances in Management and Technology (iCAMT - 2013) ,* vol. iCAMT, nº 23-27, p. 975, 2013.

[25] L. Inchaiwong y A. Sajeev, "A comparison of Software Process Models," de *SERP 2008 International Conference on Software Engineering Research and Practice*, 2008.

[26] L. Inchaiwong, A. Sajeev y S. Ramingwong, "SAM: An analysis Method of Software Process Models," de *IASTED Conf. on Software Engineering*.

[27] R. Alvarado, L. Delgado y A. Dávila, "Mapeo y evaluación de la cobertura de los procesos de MPS.Br a los procesos de la categoría de Operación de MoProSoft," de *XI Simpósio Brasileiro de Qualidade de Software*, Fortaleza, 2012.

[28] C. Halvorsen y R. Conradi, "A taxonomy for SPI Frameworks," de *Proc. 24th NASA Software Engineering*, 1999.

[29] C. Halvorsen y R. Conradi, "A taxonomy to compare SPI Frameworks," de *EWSPT '01 Proceedings of the 8th European Workshop on Software Process Technology*, London, 2001.

[30] M. Zarour, J. Desharnais y A. Abran, "A Framework to Compare Software Process Assessment Methods Dedicated to Small and Very Small Organizations," de *International Conference on the Software Quality - ICSQ'07*, 2007.

[31] M. Paulk, "A taxonomy for improvement frameworks," de *Fourth World Congress for Software Quality*, Bethesda, Maryland, 2008.

[32] L. Wen y T. Rout, "Using Composition Trees to Validate an Entry Profile of Software Engineering Lifecycle Profiles for Very Small Entities," de *12th International Conference, SPICE 2012*, Spain, 2012.