Using Qualitative Metasummary to Synthesize Empirical Findings in Literature Reviews

Danilo Monteiro Ribeiro, Marcos Cardoso
Center of Informatics
Federal University of Pernambuco
Recife, Brazil
{dmr, mjmcj}@cin.ufpe.br

Fabio Q. B. da Silva
Centre for Informatics
Federal University of Pernambuco
Recife-PE, Brazil
+55 (81) 21268430

Centre for Advanced Studies and Systems at Recife – C.E.S.A.R
Recife-PE, Brazil
+55 81 3425.4700
fabio@cin.ufpe.br

César França
Centre for Advanced Studies and Systems at Recife – C.E.S.A.R
Recife-PE, Brazil
+55 81 3425.4700
franca@cesar.org.br

ABSTRACT

Context- A common problem in Systematic Reviews in software engineering is that they provide very limited syntheses. **Goal**- In the search for alternatives of effective methods for synthesizing empirical evidence, in this paper, we explore the use of the Qualitative Metasummary method, which is a quantitatively oriented aggregation of mixed research findings**. Method** - We describe the use of qualitative metasummary through an example using 15 studies addressing antecedents of performance of software development teams. Qualitative metasummary includes extraction and grouping of findings, and calculation of frequency and intensity effect sizes. **Results** – The instance described in this paper produced a 10-factor model that effectively summarizes the current empirical knowledge on performance of software development teams. Then, we assessed the method in terms of ease of use, usefulness and reliability of results. **Conclusion** – The Qualitative Metasummary method offers rich indexes of experiences and events under investigation, focusing on the effects of a variable over other, which is consistent with the central interest of systematic reviews. However, its main limitations are (i) challenging comparability/integratability between primary studies, (ii) loss of detailed contextual information, (iii) and the great deal of effort demanded to synthesize larger sets of papers.

Categories and Subject Descriptors

D.2.9 [**Management**]: Programming Teams

General Terms

Experimentation, Human Factors, Performance

Keywords

Research synthesis; Systematic review, Qualitative metasummary, Team performance.

1. INTRODUCTION

Systematic Reviews can exert an important role in the advance of knowledge because they are aimed at generating synthesized information on a given field or research topic [1]. However, although several systematic reviews in software engineering have largely mapped key research areas and identified open research questions in recent years, there are some limitations with its current use. A common critique to this type of review is that the syntheses of primary studies are often poorly conducted [1][2]. However, full interpretive syntheses, such as those using meta-ethnography, require time, maturity of research team, experience with the methodology itself, and deep understanding of its philosophical stance [3].

Qualitative metasummary, which figures among the known techniques of research syntheses, can be useful to bridge a link between superficial mapping studies and deeper interpretative syntheses of mixed studies [4]. In this article, we describe a worked example of qualitative metasummary to synthesize a set of 15 empirical studies focusing on antecedents of performance of software engineering teams. In a longer term, our aim is to conduct a comparative evaluation of alternative methods of synthesis to help software engineering researchers to enhance the quality of their systematic literature reviews.

Our research group is mainly interested in human aspects of the software development, and has conducted systematic reviews in areas such as personality of programmers [12], motivation of software engineers [13], distributed software development [14]. Thus, we decided to run this worked example on software development teams performance. Thus, although the results of this specific example are somewhat not surprising, the main contribution of this article is showing how the qualitative metasummary helped this synthesis to go beyond a description of a set of research topics.

This paper is organized as follow: in Section 2, the steps for the Qualitative metasummary are explained in details. Section 3 describes the execution of a worked example, whose results are discussed in Section 4, along with lessons learned from this experience. Finally, Section 5 provides some concluding remarks.

2. METHOD

In 2007, Sandelowski and Barroso [10] proposed the Qualitative metasummary method as a quantitatively oriented aggregative study, aimed at finding and exposing patterns of findings from mixed-method research. It was proposed as an intermediary step between large literature surveys and more interpretive integrative studies. It is organized basically in four steps: extracting of

ESEM'14, September 18–19, 2014, Torino, Italy.

http://dx.doi.org/10.1145/2652524.2652562

findings (see Section 2.1), grouping findings (see Section 2.2), abstracting findings (see Section 2.3) and calculating frequency and intensity effect sizes (see Section (2.4)

2.1 Extracting Findings

This step consists of identifying what the actual findings of the primary studies are. Sandelowski and Barroso[10] recommend that, to begin the extraction process, the researcher must have a definition of the findings of interest, because it helps to identify what material to extract, and to maintain the whole process consistent. Indeed, the researchers may need to review, to expand or limit, their definition of finding, but every time that it happens, they must return to those reports extracted under previous definition, and review the findings against the new definition [11].

2.2 Grouping Findings

In the grouping process, the researchers will assemble groups of findings that seem to deal with similar topics, just as in a regular *in-vivo* coding process. In this step, the researchers achieve a sense of the variety of topics covered in the evidence presented in all primary studies [11]. Sandelowski and Barroso [11] suggest that the grouping process must primarily preserve the meaning and the complexity of the findings as given in the original reports, to optimize the validity of the groups [5]. The grouping process also enables the researchers to identify the relationships that exist between the concepts investigated in the primary studies and the phenomenon of interest.

2.3 Abstracting Findings

In this step, after all of the relevant findings are grouped, the researcher must carefully assign their labels, in order to make them as accessible as possible to the readers, without prejudice to their original meanings. Sandelowski and Barroso [10] propose the creation of files, in which the researchers can see all topically similar findings together. Then, the researchers could work in each file, move back and forth between the edited statements of topically similar extracted findings, and develop statements of abstracted findings until they have a set of statements that concisely, but comprehensively, captures the content of those findings, preserving the context in which they appeared. According to Sandelowski and Barroso [10], when the researchers organize the findings as such, new findings that apparently were not there and that could be expected theoretically or logically may arise.

2.4 Calculating frequency and intensity of effects

Onwuegbuzie [7] believes that to assess the relative magnitude of the abstracted results, researchers should estimate their *frequency effect sizes*. The frequency effect sizes are computed by taking the number of studies containing a finding (ignoring reports derived from common parent studies that present the same data and findings) and dividing it by the total number of studies [11]. The frequency effects size communicates how much each abstract finding is recurrent within the total universe of articles, leading researchers to identify patterns and conflicts among the primary studies.

To ascertain which reports contributed the most to the set of findings, the researchers must calculate their *intensity effect size* [10]. The intensity effect size takes the number of findings in each study and divides it by the total number of findings across all studies. This index also communicates the focus of each study, and can be useful to track possible connections between abstract findings.

3. RESULTS

This study began with a set of studies, comprising 281 papers, from the union of all papers found in the systematic reviews developed by our research group. Further details on those systematic reviews are out of the scope of this article, and should be checked out in the references [12][13][14]. Titles and abstracts were read and a regular process of paper selection was carried out. Since each review had its own inclusion and exclusion criteria regarding the relevance of the papers, we added only one extra criterion: *studies must have performance of software engineering teams as unity of analysis*. We found 15 papers in our references matching this criterion, which then accounted as the input for our qualitative metasummary process. Overview information about the studies, such as year of publication and most frequent authors, methods and publishers, are ignored in this paper in favor of the synthesis process, because of the short page limit.

3.1 Extracting and Grouping findings

As recommended by Sandelowski and Barroso[10], we first set up our definition of finding, meaning *any interpretive claim, relating whatever concept to performance of software engineering teams, made in the primary studies based on, and supported by, data presented in the own paper.* Then, the 15 studies were thoroughly read in order to extract their findings. Our definition of finding did not change during this process. Forty one (41) findings were extracted, and Table 1 shows a subset of the findings, as examples. Findings were, then, grouped with respect to their antecedent variable of interest, as exemplified in Table 1.

3.2 Abstracting the findings

We abstracted the findings and edited the labels with care to avoid bias and to preserve their meaning. The result of this process is portrayed in Table 2.

3.3 Frequency and intensity effect sizes

Finally, we calculated the frequency and the intensity of the effect sizes, as shown in Table 3.

3.4 Interpretation of the data

The data presented in the previous subsections indicates that there are only few studies addressing antecedents of performance of software engineering teams. Nonetheless, these few studies witness that performance depends on a complex set of factors that are interdependent and need to be considered together. Nine different concepts have been related to team performance in software engineering: personality of the team members, task characteristics, group cohesion, group conflict, group spontaneity, members' Motivation, and other characteristics of the team members, of the teams and of the organization. However, only six of them have been investigated more than once.

Notice that personality of team members is the most frequent investigated concept, and that four out of five studies support its influence on team performance, which strengthens the evidence. However, as discussed in PS04, there are different approaches for studying the influence of personality in software teams, such as looking only at the best fit for specific roles (team-leader, analysts, programmers), or looking at the team as a whole combination of different people. Although the four articles agree that personality influences performance, PS04 casts doubt in the group heterogeneity effect.

Our data also reveals interesting insights about Cohesion of software teams. While PS05, PS07 and PS09 evidence the relationship between cohesion and performance, two other studies dispute this fact (PS06, PS11). Furthermore, we uncovered the fact that software engineering research have so far deposited some effort to understand mainly elements of structure (G3, G6, G7) and composition (G1, G2, G4, G5, G9) of teams. Future research efforts shed some light on team processes [15] in software engineering. The main limitation of this study is that, by

abstracting findings and grouping them into similar categories, we may have accounted distinct constructs for apparently similar concepts. Moreover, adding a primary study quality assessment to this process could significantly improve the meaning of these results.

Table 1 - Examples of findings

Finding	Group
Finding 01. "This pilot study suggests that personality diversity is a strong predictor of success" [PS03]	G1. Personality
Finding 08. "The case example of IS development teams presented here suggests that personality types are an important factor in successful team performance" [PS09]	
Finding 10. "Our results did not show any significant impact of personality heterogeneity among team members on team performance" [PS04]	
Finding 18. "Group cohesion was positively related to overall performance" [PS07]	G3. Cohesion
Finding 19. "The results indicate that it [Cohesiveness] had the largest influence on performance" [PS10]	

Table 2 – Abstract findings, references and frequencies

Group (G)	Abstract Finding (A)	References
G1. Personality	(A1)…affects team performance	PS02, PS03, PS04, PS09
	(A2)…has a non-significant impact on team performance	PS04
G2. Tasks	(A3)…affect team performance	PS06, PS08, PS14
	(A4)…do not affect team performance	PS13
G3. Cohesion	(A5)… affects team performance	PS05, PS07, PS10
	(A6)… has a non-significant impact team performance	PS11
	(A7)… do not affect team performance	PS06
G4. Motivation	(A8)…affects team performance	PS12
G5. Member characteristics	(A9)…affect team performance	PS10, PS11, PS15
G6. Team characteristics	(A10)…affect team performance	PS01, PS07, PS11
G7. Organization characteristics	(A11)…affect team performance	PS07
G8. Conflict	(A12)…was not significantly related to team performance	PS07
G9. Group spontaneity	(AF13)…has a positive relationship with team performance	PS05
	(A14)…has a negative relationship with team performance	PS05

Table 3 - Frequency and Intensity effect sizes

Group	Abstract Finding	[PS01]	[PS02]	[PS03]	[PS04]	[PS05]	[PS06]	[PS07]	[PS08]	[PS09]	[PS10]	[PS11]	[PS12]	[PS13]	[PS14]	[PS15]	**Frequency**
G1	A1		1	1	6					1							4/15
	A2				1												1/15
G2	A3						2		1						1		3/15
	A4													1			1/15
G3	A5					1		1			1						3/15
	A6											1					1/15
	A7						1										1/15
G4	A8	1											5				2/15
G5	A9										2	1				1	3/15
G6	A10	5						1				1					3/15
G7	A11							1									1/15
G8	A12							1									1/15
G9	A13					1											1/15
	A14					1											1/15
Intensity		6/41	1/41	1/41	7/41	3/41	3/41	4/41	1/41	1/41	3/41	3/41	5/41	1/41	1/41	1/41	

4. DISCUSSION

Qualitative Metasummary is a synthesis method supposed to offer a rich index of research on a specific field of interest. In the following paragraphs, we present our evaluation of this method in terms of its ease of use, usefulness and reliability of results, based on our experience on carrying it out, described in Section 3.

Regarding its ***ease of use***, the qualitative metasummary is straightforward and cost-effective. There is enough documentation available about this method [6, 8, 9, 10, 11], but we were not able to find informative examples of its use in software engineering, which is our prime goal in this article. Although we believe that this goal was achieved, it is still not clear how the calculus of *intensity effect size* adds to its result. Considering that we are aggregating mixed-method studies, and that we are not saying anything about the quality of the primary studies, it does not seem reasonable to claim that one study or another contributed the most in this topic based only on their *intensity effect size* index.

Regarding its ***usefulness***, we evidence that the Qualitative metasummary produces results very well connected to the findings from primary studies. The fact that it deals with mixed-methods studies is also useful for the software engineering field [1]. Regarding its ***reliability***, the Qualitative metasummary produces transparent and auditable data. Although the findings are abstracted from their original context, it keeps track of all the process. In the software engineering field, access to the raw data can be a particular challenge.

Its interpretive approach also requires some seniority of the reseachers. In this work, for example, we faced different concepts of *performance* in the primary studies (using keywords such as team succes, and effectiveness), which difficulted the decision of what could (and what could not) be included. In addition, PS04 and PS05 apparently present inconsistent evidence, but a careful examination reveal that their conflicting data deal with different facets of the phenomena of interest. Thus we suggest that full reports of Qualitative Metasummaries also include its system of concepts, and explicit definitions for every group of abstract findings. Then, incompatible constructs could be treated separately.

Another limitation of the Qualitative measummary method is that it does not integrate the findings, because the limited integratability of data with mixed nature. Therefore, the *effect sizes* actually do not communicate strength of the effects. Although these indexes uncover literature conflicts and indicate interesting opportunities for future research, this method alone is not able to underpin or generate complete theories, so it needs to be composed with other more powerful methods. We suggest that this analysis could be augmented with information from the quality assessment of primary studies, and that this method could serve as an intermediary step between a more superficial review, like systematic mapping studies and a more sophisticated synthesis, like Systematic reviews [3].

5. CONCLUSION

Our purpose in this paper was to illustrate the use of a Qualitative metasummary method in a literature review, to synthesize empirical studies with different methodological approaches (qualitative and quantitative). The abbreviated data present are aimed at showing its application, and indicates that the Qualitative metasummary can be helpful to enhance the practice of systematic reviews in the software engineering field. This is the central contribution of this paper.

The main strength of the method presented in this article is that it produces a synthesis highly connected to the actual findings from the primary studies. Its main limitation, on the other hand, is the challenging comparability and integratability between primary studies. The idea of calculating "frequency" and "effect size" using such simple mathematical formulas also seems too

simplistic. In addition, researchers must be careful with the process of abstraction and editing findings to avoid loss of detailed contextual and important information.

In this paper, we present recommendations to adapt the Qualitative metasummary to the software engineering context. First, we question the utility of the calculus of intensity effect size. Second, we suggest that this method could be augmented with quality assessment data, common in systematic reviews. Qualitative metasummary can also serve as an intermediary research step, between a more superficial review and a deeper aggregative review. Meta-ethnographies [2], for example, are more suitable for the synthesis of a small set of consistently comparable studies, with respect to the objectives, conceptual background and contexts. In future works, the selection of this small set of studies can be based on the output of qualitative metasummaries effort.

Finally, this study is an experience report, and its validity may be limited by the fact that the individuals that applied the method were the same evaluating it. In the future, we plan to evolve this research by experiencing and evaluating other synthesis methods, such as the qualitative meta-synthesis [10], to construct a full comparative index to help researchers to choose the best suitable synthesis methods for their literature reviews.

6. ACKNOWLEDGMENTS

Fabio Q. B. da Silva holds a research grant from the Brazilian National Research Council (CNPq), process #314523/2009-0. Danilo Monteiro Ribeiro receives a scholarship from CNPq, process #132554/2013-5.

7. REFERENCES

[1] Cruzes, Daniela and Dyba, Tore. Research synthesis in software engineering: A tertiary study. Information and Software Technology (2011), 440-455.

[2] da Silva, Fabio, Cruz, Shirley, Gouveia, Tatiana, and Capretz, Luiz Fernando. Using meta-ethnography to synthesize research: A worked example of the relations between personality. In Empirical Software Engineering and Measurement (Baltimore, MD 2013).

[3] da Silva, Fabio Q. B., Santos, André L. M., Soares, Sérgio, França, A. César C., Monteiro, Cleviton V. F., and Maciel, Felipe F. Six years of systematic literature reviews in software engineering: An updated tertiary study. Information and Software Technology, 53, 9 (2011), 899-913.

[4] Kitchenham, Barbara Ann, Charters, Stuart, Budgen, David et al. Guidelines for performing Systematic Literature Reviews in Software Engineering. Keele, Staffordshire, UK, 2007.

[5] Maxwell, JA. Maxwell JA. Understanding and validity in qualitative research.. Harvard Educational Review, 1992.

[6] M, Sandelowsk, J, Barroso, and C, Voils. Using qualitative meta-summary to synthesise qualitative and quantitative descriptive findings. Research in Nursing & Health, 30 (2007), 99-111.

[7] Onwuegbuzie, AJ and C, Teddlie. A framework for analyzing data in mixed methods research. In Tashakkori A, Teddlie C, editors, ed., Handbook of mixed methods in social and behavioral research. Thousand Oaks, CA: Sage, 2003.

[8] Sandelowski M, Barroso J 2003 Jul and 13(6):781-820. Writing the proposal for a qualitative research methodology project.. Qual Health Res., 6, 13 (Jul 2003), 781-820.

[9] Sandelowski, M and Barroso, J. Classifying the findings in qualitative studies. s. 2003 Sep; 13(7):905-23. Qual Health Res, 905, 23 (Sep 2003), 13.

[10] Sandelowski, M and Barroso, J. Handbook for Synthesizing Qualitative Research. Springer, New York, 2006.

[11] Sandelowski, M, Barroso, J, and Voils, C. Using Qualitative Metasummary to Synthesize Qualitative and Quantitative Descriptive Findings. Res Nurs Health. 2007 February, 1, 30 (fEB 2007), 99-111.

[12] Shirley Cruz ; Fabio Q. B. da Silva ; Cleviton V. F. Monteiro ; Pedro C. F. Santos ; Isabella R. M. dos Santos . Personality in software engineering: Preliminary findings from a systematic literature review. In: 15th International Conference on Evaluation and Assessment in Software Engineering, 2011, Durham. Proceedings of the 15th International Conference on Evaluation and Assessment in Software Engineering. UK: Institute of Engineering and Technology, 2011. p. 1-10.

[13] A. César C. França ; Tatiana B. Gouveia ; Pedro C. F. Santos ; Célio A. Santana ; Fabio Q. B. da Silva . Motivation in Software Engineering: a Systematic Review Update. In: 15th International Conference on Evaluation and Assessment in Software Engineering, 2011, Durham. Proceedings of the 15th International Conference on Evaluation and Assessment in Software Engineering, 2011.

[14] Fabio Q. B. da Silva ; Catarina Costa ; A. César C. França ; Rafael Prikladnicki . Challenges and Solutions in Distributed Software Development Project Management: a Systematic Literature Review. In: 5th IEEE International Conference on Global Software Engineering, 2010, Princeton. Proceedings of the 5th IEEE International Conference on Global Software Engineering, 2010. p. 87-96.

[15] Marks, M., & Mathieu, J. (2001). A temporally based framework and taxonomy of team processes. Academy of Management Review, 26(3), 356–376.

Appendix A – Selected Papers

[PS01] Demirors, Elif, Sarmagik, Gamze, and Demirors, Onur. The Role of Teamwork in Software Development Microsoft Case Study. EUROMICRO (1997), page 129-133.

[PS02] Rajendran, Mazhil. Analysis of team effectiveness in software development teams working on hardware and software environments using Belbin Self-perception Inventory. Journalof Management Development, 24, 8 (2005), 738 - 753.

[PS03] Pieterse, Vreda, Kourie, Derrick, and Sonnekus, Inge. Software Engineering Team Diversity and Performance. SAICSIT (2006), 180 –186.

[PS04] Gorla, Narasimhaiah and Lam, Yan Wah. Who Should Work with Whom?.CommunicationsoftheACM, 47, 6 (2004), 79-82.

[PS05] Webster, Jane. Group Spontaneity. SIGCPR, 259 -267.

[PS06] Jones, Mary and Harrison, Allison. IS project team performance: An empirical assessment. Information&Management , 31 (1996), 57--65.

[PS07] Yanga, Heng-Li and Tang, Jih-Hsin. Team structure and team performance in IS development: a social network perspective. Information& Management, 41 (2004), 335-349

[PS08] Hoeg, Martin andGemuenden, Hans Georg. Teamwork Quality and the Success of Innovative Projects: A Theoretical Concept and Empirical Evidence. Organization Science, 12, 4 (2001), 435-449

[PS09] Bradley, John H. and Hebert, Frederic J. The effect of personality type on team performance. Journalof Management Development, 16, 5 (1997)

[PS10] Lakhanpal, B. Understanding the factors influencing the performance of software development groups: an exploratory group-level analysis. Information and Software Technology (1993), 486-473.

[PS11] Hasan, Bassam and Ali, Jafar. An Empirical Examination of Factors affecting group effectiveness in information systems projects. Decision Sciences Journal of Innovative Education, 5, 2 (2007), 229 – 243

[PS12] Baddoo, Nathan, Hall, Tracy, and Jagielska, Dorota. Software Developer Motivation in a High Maturity Company: a Case Study. software process improvement and practice, 11 (2006), 219–228.

[PS13] Hoegl, Martin, Parboteeah, K. Praveen, and Gemuenden, Hans Georg. When teamwork really matters: task innovativeness as a moderator of the teamwork–performance relationship in software development projects. J. Eng. Technol. Manage, 20 (2003), 281-302.

[PS14] Bahli, Bouchaib and Büyükkurt, MeralDemirbağ. Group Performance in Information Systems Project Groups: An Empirical Study. Journal of Information Technology Education, 4 (2005), 97-113.

[PS15] Hoegl, Martin and Parboteeah, Praveen. Creativity in innovative projects: How teamwork matters. J. Eng. Technol. Manage., 24 (2007), 148-166