2015 41st Euromicro Conference on Software Engineering and Advanced Applications

# What do we know about high performance teams in software engineering? Results from a systematic literature review

Alessandra C. S. Dutra, Rafael Prikladnicki
Faculdade de Informática (FACIN)
Pontifícia Universidade Católica do RS (PUCRS)
Porto Alegre, RS, Brazil
alessandra.dutra, rafaelp@pucrs.br

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

***Abstract* — A high performance team is one that exceeds all reasonable expectations and produces extraordinary results. In this work, we are interested in understanding contexts and conditions in which software engineering teams are likely to achieve this status. To this end, we are carrying out a systematic literature review to identify what are the known factors that booster or hinder the performance of software engineering teams, and what evidence is available to support such factors. In this preliminary effort, we selected 15 papers from SCOPUS, analyzed them following a qualitative meta-summary procedure. As a result, we present a summary of characteristics that positively and negatively influence different process of the teamwork, based on evidence available in the literature. This work constitutes a key preliminary result towards the design of more elaborate models and theories to predict and explain the performance of software engineering teams.**

***Keywords— software teams, high performance teams, systematic literature review.***

## I. INTRODUCTION

Reported statistics [18] with a sample of 10,000 projects around the world, revealed that the Information Technology (IT) still has issues; although 37% of the IT projects have been successful, being delivered before the deadline and within the estimated cost; 42% of the IT projects were delivered after the deadline, more expensive than the estimated, or with fewer resources that was agreed; and 21% of the IT projects were total failures, being cancelled before the delivered time, or were delivered but never used. Without a doubt, the success of software projects has become an important topic for both practitioners and researchers. However, achieving competitiveness without caring for a high qualified and high performing software development teams may be quite unlikely.

According to Faraj [7], improving the productivity and quality of projects is important, as initial approaches were focused on discovering better methodologies and tools, there is an increasing perception that the projects are characterized by challenges to communication, coordination, learning, negotiation, diversity and on how to form high performance teams for software development projects.

Previous studies indicate that the characteristics of the job under which software engineering teams work challenge the previously known theories that explain teamwork, such as Hackman's [23]. In this paper, we set out to investigate what studies are available in the software engineering literature reporting actual evidence on factors that influence the performance of software development teams. We report here the results of a systematic literature review, which selected 15 relevant papers from SCOPUS, analysed them following a qualitative meta-summary procedure. As a result, we summarize a set of known factors that positively and negatively influence different process of the teamwork, based on evidence available in the literature. We guided our analysis with Marks *et al.*.[11] team performance framework.

Among other findings that are described in the proper sections, this study shows that it is still difficult to draw suggestions on how to improve the performance of software teams purely based on the available evidence, because the studies frequently do not provide enough characterization of the studied contexts. Furthermore, we found that research focusing on interpersonal processes are more frequent than research focusing on other teamwork processes, which is consistent with other previous studies [28]. Thus, although the number of studies in this field seem to be increasing in the recent years, there is still much to investigate and learn about software engineering teams. In the end of the article, we provide recommendations on how the research on this topic could improve in the future.

This paper is divided into six sections. In Section II we present the theoretical foundations. Section III details the systematic literature review method. Section IV shows the results and discussions. Finally, in Section V, the conclusions and future work are addressed.

## II. BACKGROUND

### A. High Performance Teams

High-performance teams are formed by groups who rely on each other, base their actions on a common vision,

978-1-4673-7585-6/15 $31.00 © 2015 IEEE
DOI 10.1109/SEAA.2015.24

develop their activities through open communication, and have shared leadership, build confidence, enabling innovation from individual differences [20]. A high performance team is a group that brings together members committed to the mutual growth and personal success.

The main high performance teams attributes, according to Chiavenato [2] are: participation, accountability, clarity, interaction, flexibility, focus, creativity and quickness. Roda [21] presents a model of three levels for self-organizing teams: creating, practicing and transcending. The high-performance teams are at the last level and are characterized by technical and behavioural excellence, practicing and experimenting challenges continuously. A high performance team must have autonomy, attitude and more productivity than a traditional team and usually have great satisfaction in the work they do.

The participation in a team, according to Cleland and Ireland, increases the commitment and the fidelity of the people, resulting in delivery of high quality, work [3].According to Moscovici, a high performance team, must have its members committed to the personal growth and success of each team member. Such a team will exceed the performance of all the other teams and achieve results above expectations [12].

According to Raj [13], it is noticed that there is a major difficulty for an organization in disseminating high performance team practices, such as work reorganization, professional involvement in decision making processes and improvement in workers's skills, despite the evidence that organizations invest in these s practices to achieve greater productivity and efficiency. Companies with significant performance standards, according to Katzembach and Smith, stimulate and support high-performing teams, helping them to establish their own goals [9].

Katzenbach and Smith [9] present some characteristics of high performance teams: "Deeply personal commitments of each one to the growth and the success of the others is what distinguish high performance teams from the majority of the existing teams. Energized by this extra sense of commitment, the high performance team typically reflects a vigorous amplification of the fundamental teams characteristics: deeper sense of purpose, more ambitious performance targets, a more complete approach, more fullness in mutual accountability, knowledge interchangeably and complementarity."

Boyett and Boyett mention some companies that have achieved great results with high performance teams. The AT&T Credit Corporation has used high performance interfunctional teams in order to improve its efficiency and service to improve its efficiency and service to the client [1].

### B. Team Processes and Emergent Sates

Previous studies [27] have pointed out several models of team effectiveness for studies of software teams. For this research, we chose Marks *et al*. [11] as theoretical basis, because it presents an easy-to-use model, drawn from previous research and theories on the dynamics of teams. This framework is composed of three main elements: (1) conceptual definitions that distinguishes properties of teams from interactive processes of teamwork; (2) a temporally based model of team processes; (3) a taxonomy of team processes dimensions. They refer to emergent states as "constructs that characterize properties of the team that are typically dynamic in nature and vary as a function of team context, inputs, processes, and outcomes" (p.357). Examples of emergent states are collective efficacy, cohesion, composition, and other meaningful states, but that do not necessarily describe or represent the interaction between the team members. In contrast, team processes are defined as "members' interdependent acts that convert inputs to outcomes through cognitive, verbal, and behavioural activities directed toward organizing taskwork to achieve collective goals" (p. 357). Team processes describe the patterns through which the members interact. Additionally, team processes are described as series of attached episodes that necessarily transform inputs in outputs overtime, and outcomes from initial episodes often become inputs for the next cycle.

This temporally based model of team performance predicts that teams are engaged in different types of tasks at different stages of the work. The following taxonomy to describe these different types of actions:

- Transition processes: refer to periods of time when teams focus primarily on evaluation and/or planning activities, such as mission alignment or strategy formulation, which will guide their accomplishment of a team goal or objective.
- Action processes: are periods of time when teams are engaged in acts that contribute directly to goal accomplishment, such as monitoring the team progress toward the goals, baking up for teammates' tasks, coordinating and tracking team resources.
- Interpersonal processes: refer to processes that teams use to manage interpersonal relationships, such as conflict management, confidence building, and affect management.

It does not suggest, directly, any factors that would influence or determine the performance of teams, but their framework has been seen as effective to embrace and analyse the complexity of actual team arrangements. In this work, we adopted Marks's taxonomy of team processes to underpin our summary of the available evidence on factors that influence team performance. Next section gives more details on how this research has been carried out.

## III. Review Protocol

### A. Research Question and Context

The research reported in this article was guided by a single research question: *what evidence is available on factors that booster or hinder the performance of software engineering teams?*

Following our pragmatic philosophical stance, the exploratory nature of this question indicated us to carry out a systematic literature review [10], the leading method for conducting evidence-based software engineering studies. Da Silva [5] warns that there are different types of systematic reviews, and those ones aimed at identifying "all research related to a specific topic" are better described as mapping studies. In this paper, the authors agreed that the summary presented stands somewhere between a systematic review and a mapping study. Since the philosophical discussion of the appropriate terminology is out of the scope of this paper, we decided to adopt the term systematic review as a more generic concept.

This article is actually part of a broader research project that aims to generate a deep understanding of high performing teams in software engineering, by revisiting the definition of high performance teams, and identifying contextual conditions and practices in which teams are likely to flourish. Thus, answering this research question is a cornerstone towards the development of comprehensive models and theories for training and developing effective software engineering teams.

### *B. Automatic Search and Selection*

We were interested in retrieving studies published in conferences or journals in the Computer Science field, related to high performance teams training, characteristics and environments. Thus, we conducted an automatic search in a scientific database. Our limited time and human resources were enough to select only one data basis. Similar to the Salleh's study [15], we chose SCOPUS because of its reputation and the greater numbers of abstracts and citations indexed.

The search string used was built with the following composition:

1. "high performance team" OR "high performance teams"
2. "performance teams" OR "team performance"
3. "teams performance" OR "high productivity team"
4. "high productivity teams" OR "good team" OR "best team" OR "team productivity"
5. "software development" OR "software engineering"

The final string received the following combination:

(1 OR 2 OR 3 OR 4) AND 5

Then, we excluded those papers that: (1) do not address software development process or software engineering, (2) do not deal with software development teams, (3) written in any language but English, and (4) are not fully available in electronic format.

The initial automatic search resulted in 112 papers. Then, by reading the title and abstract, we filtered the papers, resulting in 61 potentially relevant studies. These papers were downloaded, and we read the full texts to check their relevance. Finally, 41 papers matched our selection criteria. Then, these 41 papers were submitted to our analysis and summary process, described in the next section. We detected, though, that only 15 of these papers presented actual empirical evidence on the influence of some factor over the performance of software engineering teams.

Other systematic reviews also detected a few articles, 20 articles were identified as relevant of 366 papers [25], and 15 papers were selected of 710 studies [26].

Due to space limitation in this article, the list of the 41 papers was made available at http://goo.gl/q6wiCa.

### *C. Data Analysis and Summary*

First, we designed a questionnaire form to extract and organize the following metadata: Paper, Year, Author; Conference (where published); Type (Journal, Conference); Objective; Context; Research methods; Answers our research Question (Yes or No?).

Regarding the context in which the studies were conducted, we classified in three categories: (1) **Education**: Studies that assessed an educational tool for teaching; (2) **Practice**: studies on the practice in software development; (3) **Tools, Models, Frameworks**: Studies describing models, frameworks and support tools.

Regarding the research methods, in order to avoid interpretation biases and improve the reproducibility of this study, we noted the research methods based on what was said by the authors, instead of based on what was actually done. Thus, we classified studies in Literature Review, Experiment, Survey, and Case Study, as there were no other methods reported.

In the extraction form, we also indicated whether the study fulfilled all the selection and exclusion criteria or not. For those excluded papers, we noted the reasons for which the paper was excluded in another field.

We followed a meta-summary approach to treat the evidence from the papers, as described in Ribeiro [14]. According to Sandelowski and Barroso [16], the meta-summary method is a "quantitatively oriented aggregative study, aimed at finding and exposing patterns of findings from mixed-method research". Given a selected initial set of papers, it is conducted in five basic steps:

1. Extraction of the primary studies findings. In our research, findings of interest regarded data showing the influence, relationship or affect, of any determined factor over any sort of measure of performance of software engineering teams.
2. Grouping findings: we grouped similar factors together based on their semantics as defined in the primary studies. As any other process of coding or thematic analysis, this task may incur in inconsistencies of in evaluation biases, so we kept track of the excerpts of text that represented the concrete findings, as well as their effects, in order to assure the credibility of the results [14].
3. Abstracting findings: This step is what makes the primary studies capable of being integrated, regardless their research method. However, as briefly discussed in Ribeiro [14], the existence of

several understandings and operational definitions for the concept of team performance is a natural challenge for this type of research. In order to reduce interpretation risks, we abstracted the findings oriented by the Marks et. al[11] framework of team processes, which is detailed in Section II.

4. Calculating the frequency and intensity of effects: frequency and intensity of effects are quantitative indicators that reveal, respectively, the recurrence of determined effects, and possible connections between factors. In this study, although we report these numbers, we decided not to give much attention to these quantitative indicators because of the small amount of papers.

5. The meta-summary approach is strongly evidence-oriented. Instead of taking the interpretation of the primary study's reporters for granted, we actually must scrutinize the data presented as results of the primary studies.

## IV. Results and Discussion

### A. General View of the Papers

There has been stability in the number of papers published every year addressing factors that influence the performance of software engineering teams. In the last ten years, a balanced number of papers has been published by Journals (7/15) and Conferences (8/15). As shown in Table I, studies are more frequently carried out with practitioners (9/15), and, secondly, with students (4/15). Regarding the research methods, Case Studies (8/15) are slightly more common than Surveys (5/15) in our sample.

Rather than expected, by crossing methods with subjects (Table I), it is possible to reveal that there are three equally frequent types of studies: surveys with practitioners, case studies with practitioners, and case studies with students. The lack of intervention research studies, such as experiments and action-research, is justifiable by the fact that such type of research with human subjects is naturally challenging. Additionally, these data on research methods reveal the exploratory nature of our current research questions in this field.

Among the authors of the 15 papers, we could find only two authors that published more than one paper: Marta Hause [P10][P12], and Martin Hoegl [P1][P13]. All the other authors in our list have published only one paper. Both Hause's papers refer to the same development project. In a traditional systematic review selection process, when two papers referring to the same study are selected, usually only one of them should stay on the list, following some choice method. In this case, [P10] and [P12] discuss different chunks of data, so we decided to keep both of them.

The development processes studied are rarely detailed in our sample of studies. As shown in Table I, seven papers did not provide any detail about the development contexts that they were collecting data. Six studies addressed distributed teams ([P1][P9][P10][P11][P12][P15]) and two addressed agile co-located subjects ([P6][P8]). This table also shows that there is a concentration of case studies on distributed software development, while surveys tend provide poor information about their contexts.

### B. Grouping factors and abstracting findings

We, then, grouped the text excerpts in more abstract factors, according to their semantic similarity. As explained in Section III, we have not checked the theoretical consistency of the primary studies, and the compatibility of similar concepts adopted in different studies, because both of these tasks are out of the scope of the meta-summary process. Table II shows examples of this process in practice.

TABLE I. Methods versus Subjects

| | | Case Study | Survey | Experiment | Literature Review | # |
|---|---|---|---|---|---|---|
| Subjects | Practice (Industry professionals) | P8, P11, P13, P15 | P1, P2, P3, P14 | P5 | - | 9 |
| | Education (Students) | P4, P9, P10, P12 | - | - | - | 4 |
| | Tools, Models, Frameworks | - | P7 | - | P6 | 2 |
| Context | Agile | P8 | - | - | P6 | 2 |
| | Distributed | P9, P10, P11, P12, P15 | P1 | - | - | 6 |
| | Not Specified | P4, P13 | P2, P3, P7, P14 | P5 | - | 7 |
| # Number of studies: | | 8 | 5 | 1 | 1 | |

TABLE II. Instance of the Grouping and Abstraction Process

| Evidence | Factor | Finding |
|---|---|---|
| Finding 13. **Autonomy** generally increases response efficiency.[P8] | Autonomy | positively influences performance |
| Finding 14. There were also differences in the **types of decisions made** by the high and low performing groups especially in terms of decisions during the software development lifecycle. [P12] | | positively influences performance |
| Finding 58. **Diversity** slows down team response due to conflicts and costly communication.[P8] | Diversity | negatively influences performance |
| Finding 60. **team diversity**, was also found to have significant influences on speed to market and the functionality of the new software product. [P2] | | positively influences performance |

We have noticed, though, that the primary studies presented different units of analysis. For example, Personality as discussed in [P5] and [P7] represents an individual characteristic; while Diversity (of personal profiles), as discussed in [P2], represents an attribute of the team.

Therefore, as the analysis progressed, and in order to keep the coherence, we categorized the factors according to

their original paper´s unit of analysis, which could be one of the following three types:

- Team emergent states: consistently with Marks *et al*. [11], team emergent states refer to meaningful attributes of teams that do not represent interactions of team members, but are rather shown to moderate the performance of team processes;
- Individual characteristics: rather than representing characteristics of the teams, it represents characteristics of the team members that moderate the performance of teams. These attributes do not necessarily describe attributes of the team as a whole;
- Support tasks: refer to patterns of interaction between team members that are not directly covered by Marks *et al*..[11] framework and taxonomy, but are shown to moderate the team processes.

Table III lists the factors categorized in each group. The frequency of each factor is represented by the number of papers mentioning each of them. In general, the frequency is very low for all factors. Only "communication" and "motivation" have been addressed in four papers each, while "personality", "autonomy", "diversity", "mutual respect" and "team size" have been addressed in two papers. All the other factors appeared in only one paper each.

TABLE III. INFLUENTIAL FACTORS

| Team emergent states | Individual characteristics | Support Tasks |
|---|---|---|
| Managerial Involvement [P14]<br>Organizational Commitment [P3]<br>Leadership Style [P10]<br>Trust [P6]<br>Self-efficacy [P6]<br>Cohesion [P9]<br>**Autonomy [P8, P12]**<br>Shared Information[P10]<br>**Diversity [P2,P8]**<br>Empathy [P2]<br>Emotional Intelligence [P2]<br>Turnover [P11]<br>**Team Size[P11,P15]** | Intelligence [P5]<br>Learning ability [P5]<br>**Mutual respect [P6, P11]**<br>Knowledge [P7]<br>**Personality [P5, P7]**<br>**Motivation [P3, P5, P7, P11]**<br>Attitudes [P7] | Goal Setting [P13]<br>**Communication [P4,P7,P12,P10]**<br>Work breakdown [P10]<br>Work satisfaction [P6]<br>Guard Activities [P14] |

In the following subsection, we detail each individual evidence presented in all 15 primary studies, and how they influence different dimensions of teamwork processes, following the Marks *et al*..[11] taxonomy of team processes. Once more, in order to avoid interpretation problems when describing these studies, we use the same terminology as used in the text of the original paper.

### C. Team Processes Dimensions

#### 1) Transition processes

Transition processes are the periods of time when teams focus primarily planning activities. We included in this category papers addressing the following concepts: organizational alignment, managerial involvement, and goal setting. Among the selected papers, the transition processes figured as the least studied dimension of team processes. Three studies presented evidence on factors that influence the performance of the transition processes.

Chen *et al.* [P3] conducted a Survey in industry with 65 IS managers and IS professionals and its results suggest that project teams with high levels of organizational commitment are more likely to have better project performance. Guinan *et al.* [P14] carried out an examination with Survey of 66 teams from 15 companies, a total of 369 team members and 110 stakeholders and concluded that highly involved managers influence planning processes commonly associated to high-performing teams. Finally, Hoegl and Parboteeah [P13] describes a Survey with a total of 575 interviews with members, leaders, and (team external) managers referring to 145 software development teams were conducted in four German software development laboratories, showing that the way that team set their goals is positively corelated to team efficiency and effectiveness.

#### 2) Action Processes

Action processes refer to those activities that arise when teams are engaged in tasks that contribute directly to the goal accomplishment. We grouped in this category papers addressing things such as improvisation, technical knowledge and learning, autonomy, coordination, information sharing, and others. We found five papers reporting factors that positively influence the performance of action processes, and three papers reporting negative factors.

Hause [P10] conducted a Case study with students from a Swedish and American University in teams of 5-6, its results suggest that high performing teams had a leadership style that was more suitable to teamwork than the low performing teams and high performing teams were better at sharing information and had less conflict. Other results found that the high performing teams were more focused on specific tasks, had an even spread of communication, participation and work breakdown.

On the negative side, Staples and Cameron [P11] conducted a case studies of six virtual teams from three different companies, in different industries (i.e. high-tech, consulting, and manufacturing), a total of 39 team members were interviewed and concluded that the geographic dispersion of team members in virtual teams makes it a significant challenge for organizations to develop and maintain effective virtual teams.

Åmite *et al*. [P15] conducted a case study in a Northern European software organization nationally distributed across two locations and its results suggest that the team or network needs to be compact, given that mutual adjustment in its pure form requires everyone to communicate with everyone.

Georgieva *et al*.[P5] describe the used well-known methods in order to conduct the study over the personal features and their influence on the working process and they found out that the four most important factors for a

productive employee are Motivation, Conscientiousness, Intellect and Agreeableness

Dyba and Dyngsoir [P6] present a Literature Review and show that development teams have faith in their own abilities, show respect and responsibility, establish trust, and preserve the quality of working life.

Siau *et al*. [P7] Conducted a interviews with twenty-one IS professionals, all located in the USA, this researched was based on a qualitative analysis of Repertory Grid sessions. It identified a number of categories of good IS team member characteristics, namely teamwork orientation, values/attitudes, knowledge, personality, working/cognitive ability, interpersonal/ communication skills, management skills, and professional orientation. The aggregated results suggest that working/cognitive ability, attitude/motivation, knowledge and interpersonal/communication skills are perceived to be the most important characteristics of good IS development team members.

We found no paper reporting negative effects of individual characteristics over the performance of action processes in software teams.

Jiang *et al*. [P4] describes a Case Study with seven student teams enrolled in the Software Engineering Group Project course in semester 2, 2010, at the University of Adelaide, Australia. The research concluded that the student team tends to perform better if team members communicate more to share knowledge, discuss about using resources, coordinate tasks effectively, and help each other frequently.

Dyba and Dyngsoir [P6] point out that preserving the quality of working life was observed through constructive discussions in the planning game, taking into account the needs of individuals in pair programming, and adhering to 40 hours-work weeks. In addition, one team took regular breaks and identified several ways to relieve developers in hectic periods.

On the other hand, the survey presented by Guinan *et al*. in [P14] found that guard activities are negatively related performance. Guard activities are activities designed to keep information inside the team until the team desires to release the information. Guard activities monitor and restrict the teams' external influences.

*3) Interpersonal Processes*

Interpersonal processes are those ones used to maintain interpersonal relationships in the team. In this group, we put together aspects like mutual trust and respect, self-efficacy, communication, empathy and other factors. In this study, the highest rate of papers reported factors affecting interpersonal processes.

Dyba and Dyngsoir [P6] shows that respect for one's team members and a sense of responsibility were manifested via the way in which work was assigned; active agreement was required. Individuals clearly felt that they had the respect of their fellow team members and were therefore empowered to take on responsibility in this way.

Staples and Cameron [P11] suggests that respect for one's team members and a sense of responsibility were manifested via the way in which work was assigned; active agreement was required. Individuals clearly felt that they had the respect of their fellow team members and were therefore empowered to take on responsibility in this way. Other results showing that people who felt their team had good performance felt that there was a strong team spirit and were much more likely to identify innovations that had occurred to make the team more effective and a team member who had low motivation also perceived that the team's spirit was low. Also, people who had lower satisfaction with their team felt that there was not a strong team spirit and often felt that coordination could be improved.

Swigger *et al*. [P9] conducted a case studies from two pilot projects involving 152 students from the US, Panama, UK, and Turkey and concluded that the collaborative work factors of cohesion, team atmosphere, and support are considered to be important for groups to work successfully and perform the necessary tasks. It also suggests that cohesion is one of the more important elements in a high performing team, not unlike what has been found in other studies

Hause [P10] reported that High performing teams had a leadership style that was more suitable to teamwork than the low performing teams.

Günsel and Açikgöz [P2] conducted a Survey in industry with 86 software development projects in Istanbul that have affiliations with European and American firms and its results suggest that aspects such as mutual understanding, emotional intelligence, empathy, autonomy, diversity, synergy have positive effects on several market variables (e.g. time-to-market and functionality).

Hause *et al*. [P12] describes a Case study with 2 universities', Uppsala University (UU) in Sweden and Grand Valley State University (GVSU) in the USA. There were 16 teams in total, 13 teams of 6 students (three from each university) and three teams of 5 students in each team. The Swedish students were in their third year of university study and American students were in their third or fourth year. Hause *et al*. [P12] concluded that:

- There were also differences in the types of decisions made by the high and low performing groups especially in terms of decisions during the software development lifecycle.
- One of the differences between high and low performing groups is the amount of communication produced. The low performing groups have more communication than the high performing groups. Analysis of their work process suggests that it is not the quantity of the communication but the quality that is important in determining performance.
- The project was the same for all the teams, however all teams had individuals with different backgrounds and experiences, which made their work process unique. The differences between the high and low performing teams was due to the quality of the

communication and the process and timing of specific actions.

Georgieva *et al*. [P5] also corroborates the influence of motivation and individual personality traits over the performance of interpersonal processes.

Siau *et al*. [P7] shows that working/cognitive ability, attitude/motivation, knowledge and interpersonal/ communication skills are perceived to be the most important characteristics of good IS development team members. Based on a qualitative analysis of raw RepGrids, they developed a number of categories of good IS team member characteristics, namely teamwork orientation, values/attitudes, knowledge, personality, working/cognitive ability, interpersonal/ communication skills, management skills, and professional orientation.

Chen *et al*. [P3] show that the relationship between project team characteristics and performance can be mediated by team motivation. Management should pay more attention to team motivation within project teams so that a better project performance can be achieved.

Å mite *et al*. [P15] also shows how the team size negatively affect the performance of interpersonal processes.

## V. Discussion And Conclusion

### A. Main findings and implications

As a result of this study, we found 15 papers reporting factors that influence on different dimensions of team processes in software engineering teams. Our main contribution was the analysis and summary following a systematic procedure, organizing the results according to a reasonable theoretical framework of understanding teamwork, which helped us to identify research gaps and opportunities to improve our understanding in this specific topic. These results are summarized in Table IV.

In this study, we could evidence that most of the empirical work that has been done in this field assume an exploratory approach, which may mean that we are still starting to understand the problems related to teams in software engineering. These primary studies deal with apparently distinct types of teams (e.g. virtual/distributed, large and small, industrial and educational) also looking at different aspects of performance. We believe that this integration may serve as an initial effort towards the development of theoretical frameworks to underpin empirical studies in this field, hence enabling some knowledge accumulation.

We could also evidence on this study the lack of information about the context on which the studied teams are embedded, which has been corroborated by previous secondary integrative studies in software engineering (e.g. [14][27]. Another noticeable phenomenon is that, as reported in Section III (B), 26 papers were excluded from our analysis because we did not find empirical content regarding the teamwork factors. Therefore, we claim that the quality of reports of primary research on teams in software engineering should definitely improve. A guideline to report studies of teamwork in software engineering would help to overcome this gap in the long term

TABLE IV. Summary of the studies

| | Team emergent states | Individual characteristics | Support Tasks |
|---|---|---|---|
| **Transition Processes** | Organizational Commitment (+) [P3]<br>Managerial Involvement (+) [P14] | - | Goal Setting (+) [P13] |
| **Action Processes** | Leadership Style (+) [P10]<br>Shared Information[(+) P10]<br>Turnover (-) [P11]<br>Team Size[(-) [P11,P15] | Intelligence (+)[P5]<br>Learning ability (+) [P5]<br>Mutual respect (+) [P11]<br>Knowledge (+) [P7] | Communication (+) [P4, P7]<br>Work satisfaction (+) [P6]<br>Work breakdown (+) [P10]<br>Guard Activities (-) [P14] |
| **Interpersonal Processes** | Trust (+) [P6]<br>Self-efficacy (+) [P6,P11]<br>Cohesion (+) [P9]<br>Leadership Style (+) [P10]<br>Autonomy (+) [P8,P12]<br>Diversity (+) [P2,P8]<br>Empathy (+) [P2]<br>Emotional Intelligence (+) [P2] | Personality (+) [P5,P7]<br>Motivation (+) [P3, P5,P7]<br>Motivation (-) [P11]<br>Mutual Respect (+) [P6]<br>Attitudes (+) [P7] | Communication (+) [P10,P12] |

(+) Positive influence (-) Negative influence

In addition, it is very difficult to draw specific practical recommendations to improve performance of software engineering teams from this set of studies, because we face limitations regarding not only the compatibility of the concepts but also the compatibility of contexts. Compatibility of concepts is partially addressed in the meta-summary process of synthesis, although in a very simple manner. We recommend future work to build up on our results, by conducting more powerful methods of mixed-methods syntheses, such as meta-ethnography [19].

Remarkably, the teamwork research in software engineering has been shedding much light on interpersonal processes. In contrast, in our sample, we found little evidence on factors that influence the performance of transition processes of teamwork in software engineering. In particular, we found no study interested in individual characteristics influencing the planning processes, for example.

### B. Threats to validity, and future work

Observe that the primary studies selected in our systematic review present significantly more evidence on positive than negative factors. That may represent a publication or a methodology bias. By publication bias, we mean that papers addressing positive factors may be more easily accepted in those conferences and journals that we analyzed in this study. Alternatively, researchers might be less prone to investigate those negative aspects. By methodology bias, we mean that the empirical design of primary studies of workplace behavior per se may baffle the actual influence of the studied factors, phenomenon

commonly known as the Hawthorne Effect [22]. Teamwork is actually a complex set concurring of positive and negative factors, future studies should consider treating both of these bias as threat to their validity.

Both these phenomena may also be a reflex of our limited search string, which did not include words such as "low performance" or "bad teams". Additionally, we believe that if our search string was expanded to include each factor that we have found in this study, there could be actually much more valid evidence. For example, if our search include "communication", there could be thousands of other studies. On the other hand, the analysis process would be much more time consuming. We are still to discuss how to address this issue in future studies.

In general, the 15 studies detailed in this paper represent only an apparently set of disconnected studies. It is important to notice that, given limitations of time and human resources, we decided to carry out the automatic search in only one scientific repository (SCOPUS), which may occur in other types of publication and selection bias according to Wohlin *et al*. [26]. We constantly refer to the original papers so that the readers interested in more specific factors can track the references and check these concepts. It is indeed a limitation of this article and future work must address this issue properly. One way we plan to improve our research results is by expanding and improving our search procedures to be able to draw stronger claims.

## REFERENCES


[1] Boyett, J.H; Boyett, J.T. 1998. The Guru Guide-The Best Ideas of the Top Management Thinkers . New York: Wiley.

[2] Chiavenato, I. 2008. People management: the new role of human resources in organizations. Rio de Janeiro: Elsevier, 3a edition.

[3] Cleland, D. I. ; Ireland R. L. 2000. Project Manager`s portable handbook. New York: McGraw-Hill, 257p.

[4] Cruzes, D. S., & Dybå, T. 2010. Synthesizing evidence in software engineering research. In *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '10* (p. 1). New York, New York, USA: ACM Press. DOI=10.1145/1852786.1852788

[5] Da Silva, F. Q. B., Santos, A. L. M., Soares, S., França, A. C. C., Monteiro, C. V. F., & Maciel, F. F. 2011. Six years of systematic literature reviews in software engineering: An updated tertiary study. *Information and Software Technology*, *53*(9), 899–913. DOI=10.1016/j.infsof.2011.04.004

[6] Easterbrook, S., Singer, J., Storey, M.-A., & Damian, D. 2008. Selecting empirical research methods for software engineering. In *GUIDE TO ADVANCED EMPIRICAL SOFTWARE ENGINEERING* (pp. 285–311). Springer Science + Business Media.

[7] Faraj, S., Sambamurthy, V., 2006. Leadership of information systems development projects. In: IEEE Transactions on Engineering Management.

[8] Gibbs, W. 1994. Software's Chronic Crisis, *Scientific American,* September, 86–95.

[9] Katzenbach, J. R; Smith D. K. 1993. The Wisom of Teams. Summarized by permission of Harvard Business School Press Copyright by McKinsey & Company, Inc. 275 pages

[10] Kitchenham, B. 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering, EBSE Technical Report.

[11] Marks, A. M; Mathieu E.J; Zaccaro S. 2001. A Temporally Based Framework and Taxonomy of Team Processes. In: The Academy of Management Review, Vol. 26, No. 3, pp. 356-376.

[12] Moscovici, F. 2003. Teams work right: Multiplication of Human Talent. Rio de Janeiro: José Olympio, 8a edição (in portuguese)

[13] Raj, P .P ; Baumotte A.C.T; Fonseca D.P .D; Silva, L.H.C.M. 2006. Project Human Resource Management. Rio de Janeiro: FGV – Fundação Getúlio Vargas, 2006, 180p. (in portuguese).

[14] Ribeiro, D. M., Cardoso, M., da Silva, F. Q. B., França, C. 2014. Using qualitative metasummary to synthesize empirical findings in literature reviews, Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), Torino, Italy. DOI =10.1145/2652524.2652562.

[15] Salleh, N.; Mendes, E.; Grundy, J. 2011. Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review. In:. IEEE TSE, 37(4):509–525.

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

[17] Sommerville, I. 2010. Software Engineering, 9a edition. Pearson.

[18] Standish. 2013. *Extreme CHAOS.* The Standish Group International, West Yarmouth, MA, USA. [WWW document.] URL http://goo.gl/MU8l94

[19] Noblit, G. W.; Hare, D. 1988. META-ETHNOGRAPHY: Synthesizing Qualitative Studies. Newbury Park: SAGE Publications, v. 11. ISBN 0-8039-3022-4.

[20] S. K. Parker, and P. R. Jackson. 1993. The implementation of high performance work teams, Case Studies in Organisational Behavior and Human Resource Management, 2nd Edition (pp 42-56). London: Paul Chapman Publishing.

[21] Roda, R., 2011. Self-Organizing Agile Teams: A Grounded Theory, PhD Thesis, Victoria University of Wellington.

[22] Mayo, E. 1949. Hawthorne and the Western Electric Company.

[23] Hackman, R J. 1998.Why Teams Don't WorkTheory and Research on Small Groups, Chapter 12, edited by R. Scott Tindale et al. Plenum Press, New York.

[24] Riaz, M., Mendes E., , Tempero E,. 2009. A Systematic Review of Software Maintainability Prediction and Metrics, Third International Symposium on Empirical Software Engineering and Measurement.

[25] Hossain,W.,Babar M. A.,Paik H. 2009. Using Scrum in Global Software Development: A Systematic Literature Review. Fourth IEEE International Conference on Global Software Engineering

[26] Wohlin, C., Runeson, P., Neto, P. A. D. M. S., Engström, E., Machado, I. D. C., & Almeida, E. S. De. (2013). On the Reliability of Mapping Studies in Software Engineering. Journal of Systems and Software, 1–36. doi:10.1016/j.jss.2013.04.076.

[27] Dingsøyr, T.; Dybå, T. (2012) Team Effectiveness in Software Development - Human and Cooperative Aspects in Team Effectiveness Models and Priorities for Future Studies. 5th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE). DOI 10.1109/CHASE.2012.6223016.

[28] LePine J, Piccolo R, Jackson C., Mathieu J., Saul J. (2008) A Meta-Analysys of Teamwork process: Testes of a multidimensional Model and Relationships with Team Effectivesness Criteria. Personnel Psychology, 61 – 273-307.