Investigating Cultural Dispersion: on the Role of Cultural Differences in Software Development Teams

Stefano Lambiase
slambiase@unisa.it
SeSa Lab, Department of Computer Science, University of Salerno
Fisciano (SA), Italy

ABSTRACT
Software development, inherently a social activity, involves individuals across diverse geographical and cultural settings. Despite this nature, the existing Global Software Engineering research body encounters limitations, making the achieved results challenging to use by practitioners. This Ph.D. research project seeks to overcome these constraints by crafting a theoretical framework. The framework systematically captures cultural differences, exploring their impact on various aspects of software development and delving into practitioners’ strategies for managing these influences. Additionally, the project aims to significantly contribute to the professional software development landscape by transferring research findings to practitioners through practical tools. This framework serves as an immediate application for professionals, fostering project success through heightened cultural awareness and adaptability, thereby enhancing developer well-being in inclusive and culturally diverse environments.

CCS CONCEPTS
• Software and its engineering → Software organization and properties; • Social and professional topics → Cultural characteristics; Geographic characteristics.

KEYWORDS
Global Software Engineering; Cultural Dispersion; Socio-Technical Aspects

ACM Reference Format:

1 CONTEXT AND LIMITATIONS
Software development is de facto a collaborative and human-centered activity [4, 25]; researchers in the Software Engineering community have been exploring how human and social factors—e.g., emotions and cultural background—can profoundly influence each aspect of software development. Their investigations reveal the crucial nature of managing these factors, as they wield substantial influence over the ultimate success of a software project [6, 31].

Among the various societal challenges that may potentially impact software development, culture is emerging as a critical element that demands attention among software community members throughout the development lifecycle. Culture is defined as shared motives, values, beliefs, identities, and interpretations or meanings of significant events that result from common experiences of members of collectives and are transmitted across generations [18]. Our emphasis on culture stems from a pivotal insight presented in the seminal work by Damian and Moitra [9] and by Šmite et al. [27]: culture exerts a more substantial influence on software engineering practices than initially anticipated. This underscores the primary reason why researchers in the domain of Global Software Engineering (GSE)—which pertains to the branch of software engineering aiming to establish a set of practices for managing globally distributed software teams—have extensively delved into the impact of cultural aspects on activities related to distributed software development and maintenance [7, 22, 29].

The key insight derived from these research papers is that culture can impact various aspects of software development in a manner that extends beyond observable and measurable dimensions. Despite acknowledging the considerable effort invested in constructing a substantial body of knowledge on culture in software engineering, our research identifies several fundamental limitations in the current state of the art:

Contrasting Results. Literature on culture in software development presents a contrast in terms of results. On the one hand, the presence of individuals from different cultures within the same team may lead to positive long-term effects [14, 24], such as empowering decision-making processes by broadening the spectrum of opinions. On the other hand, these differences may converge into collaboration and communication problems [5, 7, 23], potentially increasing the likelihood of interpersonal conflicts. Consequently, the role of cultural dispersion in global software engineering remains unclear and warrants further analysis.

Lack of Knowledge on Socio-Technical Aspects. Most previous studies have explored the relationship between culture and software development by concentrating on specific processes and product metrics. For example, some studies delved into understanding how culture influences the code review process [3] or code quality [1]. However, there is still a dearth of knowledge concerning how culture may influence the behavioral patterns among developers [34], particularly in terms of communication and collaboration relationships within a development community. Consequently, our understanding of the impact of culture on socio-technical dimensions remains limited.
Lack in Using Cultural Frameworks. It is noteworthy that only a limited number of studies have utilized cultural frameworks [12, 13, 16, 17], which provide (1) a set of cultural behavior of individuals associated to culture and (2) numerical values to characterize such behaviors. Moreover, previous work has predominantly treated culture as an “abstract” concept without quantifying it [22]. As a result of this approach, most of the findings reported in the literature cannot be evaluated against a reference framework, therefore hindering a more objective understanding of how culture impacts software engineering practices.

Lack of Theoretical Framework. Currently, no comprehensive theory adequately elucidates the impact of cultural differences on software development and effective strategies for managing these differences. This absence carries two significant implications: firstly, it contributes to fragmented research, frequently yielding new contributions without substantial progress in the field; secondly, it constrains the practical applicability of research findings in the software development industry, as there is no established framework readily graspable and implementable by practitioners.

Addressing the limitations above is pivotal to better understanding how cultural differences impact software development and the teams involved. Such knowledge could be bridled to empower managers to make informed decisions, consequentially increasing the success rate in software development projects and enhancing developers’ well-being.

## 2 RESEARCH OBJECTIVE

In response to the limitations above and intending to enhance the body of knowledge in Global Software Engineering, this Ph.D. research project seeks to formulate a theoretical framework. This framework is intended to encompass the influence of cultural differences (represented in a systematic form) on various aspects of software development and explore strategies employed by practitioners to managing such impacts. Given software development’s collaborative and social nature [4, 25], the guiding hypothesis posits that cultural differences significantly influence the development lifecycle and its participants, potentially influencing them positively and negatively depending on how well such “dispersion” is managed. Moreover, since software development is a socio-technical phenomenon [15, 24], a particular focus is invested in studying how socio-technical aspects are influenced by culture.

In addition to the aforementioned goal, this work aims to contribute substantially to the practitioners’ software development landscape. To achieve this, we aim to turn research findings into actionable results by creating tools that embed the new knowledge.

### Research Objective—Studying, understanding, and reporting on the role of cultural differences in the software development lifecycle, particularly focusing on socio-technical aspects and phenomena. Then, making such knowledge actionable by practitioners and researchers through the development of a theoretical framework and tools (e.g., recommendation systems) based on it.

### 3 RESEARCH PROCESS AND EVALUATION

Figure 1 reports the research method defined to answer the research questions and address the research objectives.

#### 3.1 First Objective—Main Research Activity
Concerning the first objective, three research questions have been formulated to guide the study [33].

- **RQ1**—What is the current state of the art regarding cultural aspects in software development, and what are the associated limitations in the existing research?

  The initial question aims to present the current state of the art on the primary topic under analysis and to pinpoint potential gaps that could contribute to the primary findings of the dissertation. Furthermore, scrutinizing the state of the art is not only aimed at providing new insights but also at laying the groundwork for original outcomes.

  To address the first research question, a thorough literature review of reputable sources will be conducted, seeking secondary studies (e.g., systematic literature reviews and mapping studies) on the subject. If secondary studies are not identified, the initial step of the research involves conducting a systematic literature review and a mapping study on the role of culture in software engineering.

- **RQ2**—How do cultural differences in software development teams influence the development lifecycle?

  The second research question serves as the nucleus of the Ph.D. project, encapsulating its primary objective. It is anticipated that several more specific research queries will emanate from it and from the responses to the first question, each evolving into separate endeavors. The amalgamation of these efforts is expected to culminate in a comprehensive answer to the original question.

  Concerning the second research question, the plan involves dividing the research into two phases. In the first phase, mixed-method research approaches [8] will be employed to explore culture in contexts identified as lacking contributions while addressing the first research question. In this research type, qualitative methods (e.g., qualitative analysis, grounded theory, and interviews) and quantitative investigations (e.g., statistically supported empirical studies and data mining) are conducted on the same dataset or in the same context to address similar research questions. The ultimate goal is to achieve theoretical saturation, signifying the point in category
development where no new properties, dimensions, or relationships emerge during analysis. Therefore, the theory is considered saturated if both qualitative and quantitative data converge on the same conclusions. Conversely, any disparities between the two studies would lead to the need for ulterior investigations on the matter. In light of our frequent involvement with human participants, we will prioritize ethical considerations. An ethical review board will be consulted, particularly for studies requiring ethical oversight. This ensures our research adheres to rigorous ethical standards, safeguarding participants’ well-being, rights, and privacy.

3.2 Second Objective—Technology Transfer

For the second objective, given that the primary aim is to develop tools for turning research findings into actionable results, a comprehensive literature review with multiple perspectives will be undertaken. The aim is to (1) pinpoint the optimal instrument for executing technology transfer and (2) comprehend the current state of the art regarding best practices and challenges in its development and adoption. A multivocal literature review (MLR) [10] will be conducted rather than a systematic one; MLR is ideally indicated to study aspects related to practitioners and provide results beneficial for them [10]. After establishing foundational knowledge, a series of tools will be developed and iteratively refined with the goal of offering practitioners and researchers a user-friendly and accessible means to apply insights from leading research. These tools will be presented as tool demonstration papers and disseminated through various social channels.

In terms of evaluation, a dual approach will be adopted. First, as the tools are designed for practical use, a usability evaluation involving human participants will be conducted [11]. Additionally, each tool will undergo evaluation regarding its efficacy in enabling the correct application of research findings by individuals. This evaluation will be tailored to the specific case and will encompass both human and statistical assessments.

4 EXPECTED CONTRIBUTION

The primary contributions of the research align with its objectives, encompassing two key aspects.

Theory of Dealing with Cultural Dispersion. Firstly, a theoretical framework will be developed, delving into the impact of individual culture on software development’s socio-technical aspects and how professionals navigate this dynamic. This framework, grounded in both qualitative and quantitative studies, will address limitations identified in the current state of the art and will be designed to benefit both practitioners and researchers.

Tools for Performing Technology Transfer. The second contribution involves a set of tools aimed at making research findings easily usable and widely accessible to all. Together, these contributions aim to advance understanding in the field, providing practical insights for professionals and fostering widespread dissemination of research outcomes. Moreover, the identified limitations in the state of the art will be addressed.

In order to address the limitation on socio-technical aspects, since software development is a socio-technical phenomenon [15], the primary goal is to investigate the impact of culture on socio-technical aspects (e.g., technical and social debt). For instance, a key focus of our investigation revolves around examining the influence of cultural dispersion on community smells [30], i.e., socio-technical anti-patterns within the collaborative structure of a development community that are precursors of social debt [30].

Moreover, as we deal with cultural differences, we prefer using cultural frameworks [16, 17] instead of getting stuck in the tricky
task of defining culture, which can lead to confusion. We put these frameworks into action in two ways: by talking about culture based on behaviors (the qualitative side) and by using the numbers tied to these cultural dimensions (the quantitative side). This helps us avoid being too vague in our findings.

Lastly, by conducting concurrent qualitative and quantitative experiments and comparing their outcomes, the aim is to formulate theories that comprehensively and robustly depict the current state of art and practice [8, 15, 26]. This approach seeks to (1) unveil the role of culture in software development and (2) provide theoretical frameworks usable by both professionals and researchers.

By mitigating the identified limitations, it is possible to formulate a practical, theoretical framework that delineates the influence of cultural differences on software development. This tool seeks to elevate project success rates by promoting heightened cultural awareness and adaptability, ultimately fostering improved well-being among developers in diverse environments.

5 PRELIMINARY RESULTS AND SCHEDULING

In terms of results, they are discussed in alignment with the objectives mentioned in previous sections.

Results Achieved for the Main Research. Addressing the first objective, the literature analysis identified limitations, detailed in the paper’s introduction, guiding subsequent steps and answering the first research question.

Concerning the second research question, the initial focus was on exploring how cultural differences impact the emergence of community smells and the productivity of a development community. Culture, treated through the concept of cultural dispersion, was systematically approached to provide a concrete understanding that led to three publications [19–21]:

- The investigation into the impact of culture on the social aspects of software development informed the first study. We hypothesized that cultural dispersion could influence collaboration, leading to the emergence of community smells. The study showed that cultural dispersion affects the emergence of all community smells, with nuanced results—indicating not solely negative impacts. A notable finding was the correlation between the presence of individualistic and collectivist individuals and the emergence of Lone Wolf effects. The work [21] was published at the International Conference on Software Engineering-Software Engineering in Society (ICSE-SEIS 2022).
- Expanding our focus on socio-technical metrics, we delved into the relationships between cultural dispersion and productivity.
- A mixed-method study revealed that dispersion metrics can both positively and negatively affect productivity, contingent on how managers address cultural differences. For instance, integrating individualistic and collaboration-oriented individuals might prove ineffective, underscoring the need for a nuanced approach. This work resulted in two publications [19, 20]: the first one [19] at the Software Engineering and Advanced Applications Euroconference (SEAA 2022) and the second [20] (an extension of the first) in the Journal of Systems and Software (JSS).

Moreover, in each study, we aimed to evaluate our representation of culture through cultural dispersion metrics and concepts, confirming its maturity for practical use by practitioners and researchers.

In the same context, additional work has been submitted and is currently under review to conclude the first objective through the third research question.

Results Achieved for the Technology Transfer. Regarding the second objective, we identified conversational agents, commonly known as chatbots, as a managerial tool within the context of software development. Initially, we conducted a comprehensive literature review (currently under revision) to accumulate insights into the adoption and challenges associated with these tools in the software engineering domain.

Building upon the acquired knowledge, we developed CADOCS (Conversational Agent for the Detection Of Community Smells) [32], a conversational agent published at the International Conference on Software Maintenance and Evolution (ICSME 2022). CADOCS extends a previous community smell detection tool proposed by Almarimi et al. [2], aiming to (1) enhance its usability within a well-established communication channel like Slack, and (2) augment its functionality by offering initial support for software analytics instruments crucial for diagnosing and refactoring community smells. Additionally, CADOCS has been designed for high extensibility, ensuring continuous improvement over time and seamless integration of emerging insights from cutting-edge research.

In terms of future work, as illustrated in the Gantt chart in Figure 2, our current research emphasis lies in the validation of new tools and the formulation of theories. Substantial work has been dedicated to qualitative theory development, and the construction of theory based on quantitative data (SEM) is underway, with a planned submission in the upcoming months. Following completion, a CADOCS tool update will be crafted, integrating insights derived from the conducted studies. Ultimately, the thesis writing phase is scheduled to commence in early January 2024.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RQ1—State of the art on Cultural Aspects</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RQ2—Studying the impact of Cultural Dispersion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RQ3—Representing the impact of Cultural Dispersion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 2: Technology Transfer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2.1—Tool Identification and State of the Art</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2.2—Tool Development</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2.3—Tool Evaluation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thesis Writing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: Research Activity Schedule.
REFERENCES


