Mapping Global Software Development Practices for Follow-the-Sun Process

Josiane Kroll
Computer Science School
Pontifical Catholic University of Rio Grande do Sul (PUCRS)
Porto Alegre, Brazil
josiane.kroll@acad.pucrs.br

Jorge Luis Nicolas Audy
Computer Science School
Pontifical Catholic University of Rio Grande do Sul (PUCRS)
Porto Alegre, Brazil
audy@pucrs.br

Abstract — Several organizations are developing software processes twenty-four hours, seven days per week, with geographically distributed teams. This environment of software development enables to implement the Follow-the-Sun (FTS) strategy. In this study, we perform a mapping of the literature based upon electronic searching in digital libraries to identify applied practices in development environments twenty-four hours in which can apply FTS strategy. Ours results present practices and many key aspects to FTS implementation.

Keywords - Global Software Development; Follow-the-Sun; Software Process; Software Practices.

I. INTRODUCTION

Global Software Development (GSD) is relatively new, but it has intensified over the past years [1]. Nowadays, many companies are adopting GSD because of the business globalization in software development processes that are increasing [2]. With it, companies realize the real necessity to expand their businesses for foreign markets [3] and adopting for it new software development strategies [4].

The software development globalization enables companies to create subsidiaries in other countries configuring process distribution twenty-four hours, seven days per week [5]. This GSD strategy is called Follow-the-Sun (FTS). However, the FTS implementation is difficult to achieve. It requires great effort for the teams involved [6]. If not applied correctly, it can result in failures and increase project cost [7]. In addition, many challenges are found when implementing FTS, such as, communication difficulties, coordination barriers and cultural differences [6] [8].

In the literature, few studies explore the FTS strategy and give little evidence of success. The lacking of practices and processes is creating barriers for FTS adoption and evolution in the software industry [6]. Hence, our study aims to map GSD practices to support the FTS processes. We performed a systematic mapping study (SMS) method, based upon electronic searching of main digital libraries. The SMS method provides a wide overview of the research area, to establish if research evidence exists on a topic [9].

Our work first aims to collect information about FTS. This information gives us the characterization of the FTS scenario. We then search in the literature GSD practices to support FTS process.

II. BACKGROUND AND MOTIVATION

In this section, we describe briefly the Follow-the-Sun (FTS) concept providing motivation to perform this study.

A. Follow-the-Sun Strategy

FTS is a subset of the global software development (GSD) [10]. It is characterized by software development twenty-four hours, seven days per week, with geographically distributed teams [11].

In FTS, when a team in a specific location finishes its working day, another team in a second location and different time zone takes the task over starting its working day [10]. The daily production done by a FTS team is sent to the next production site for continuation. Continuity of the working day involves exchange of task cycles between teams separated by diary handoffs [10] [12]. Handoff is a term utilized in the literature to describe the task transition process between teams. The handoff can be defined as a check-in from a work unit that will be delivered to the next production site [10].

The main goal of the FTS is to structure software development processes according to time, enabling to reduce the time-to-market [13] [14]. Carmel, Dubinsky and, Espinosa (2009) claim only this benefit.

Many companies, such as IBM and Infosys have tried to apply FTS, but abandoned it after some point, because of the difficulty of putting it into practice [11]. Nowadays we observe with great interest the software industry in FTS, but the lack of theoretical studies make the evolution of the FTS difficult [15]. Hence, we believe this is an important research topic that will help to build and design software process for FTS.
III. THE MAPPING PROCESS

We utilized the systematic mapping study (SMS) method to identify GSD practices in the literature that can be applied to FTS. An SMS is defined as a method to build a classification schema and structure the software engineering field [16]. It aims to identify all researches about a specific topic, answering questions related to evolution and trends [17]. An SMS is designed to provide a wide overview of a research area, to establish if research evidence exists on a topic [9]. We selected SMS because our goal was to explore existing studies about twenty-four hour software development. Results of this study can help to identify and to create new practices for FTS.

SMS have five steps: definition of research questions, searching for relevant papers, screening papers, key wording of abstracts, and extraction and mapping [18].

A. Definition of Research Questions

Research questions in this study were defined as follows:

RQ1: What are the characteristics of the FTS development scenario?

RQ2: Which software development practice is found in a GSD scenario for FTS?

B. Searching and Keywording Criteria

Our SMS started with the identification of keywords and search terms. We used general keywords in the search in order to identify as many relevant papers as possible. The search was conducted using the boolean search expression as follows:

('global software development' <OR> 'global software engineering' <OR> 'distributed software development' <OR> 'distributed software engineering')

The following electronic libraries were considered:

1) IEEEXplore (http://ieeexplore.ieee.org)
2) ACM Digital Library (http://www.portal.acm.or/dl.cfm)
3) Wiley InterScience (http://onlinelibrary.wiley.com/)
4) Elsevier ScienceDirect (http://www.sciencedirect.com)

We performed the search on January 2012 considering the period from 1990 to 2012, because studies in GSD began to be published in the early 1990’s [18] [19]. In Table I, we present paper number obtained in each library.

<table>
<thead>
<tr>
<th>Digital Library</th>
<th>Total results found</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEEXplore</td>
<td>555</td>
</tr>
<tr>
<td>ACM Digital Library</td>
<td>133</td>
</tr>
<tr>
<td>Elsevier ScienceDirect</td>
<td>84</td>
</tr>
<tr>
<td>Wiley InterScience</td>
<td>37</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>809</strong></td>
</tr>
</tbody>
</table>

The identification process resulted in 809 articles, excluding duplicated studies. It created the basis for the next step in our selection process.

C. Screening Papers

First, we analyzed title, abstract and keywords for each paper. If in doubt about the contribution of the paper, we read the full paper. Also, we excluded posters, panel, abstract, presentation and, articles summaries.

We still applied a set of inclusion/exclusion criteria to define practices. The criteria were:

- We only select practices recommended for FTS.
- We do not make judgment in relation to the possible applicability of the practice for FTS, when it is not described in the study.
- If the practice found is recommended for FTS, we then map it in the categories defined.

IV. RESULTS

A. Characteristics of the FTS Development Scenario (RQ1)

We analyzed definitions suggested by different authors about FTS characterization. Thus, based on the information collected, we categorized and pointed requirements of the development scenario in FTS. The result is present in Table II.

<table>
<thead>
<tr>
<th>Categorization</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teams allocation</td>
<td>Teams are geographically distributed [11] [6]</td>
</tr>
<tr>
<td>Software development strategy</td>
<td>The software product is developed on 24 hours per day [11] [20] [21].</td>
</tr>
<tr>
<td>Work collaborative</td>
<td>Teams at the end of a working day, send the work to the next team, localized in a different local, to continue the work. This process is called handoff [20] [22] [23] [24] [25] [26] [27] [28].</td>
</tr>
<tr>
<td>Software development life cycle</td>
<td>Teams depend on the handoff to continue the work [10]. At any point in time, only a local has the product [10] [11].</td>
</tr>
<tr>
<td>Number of sites</td>
<td>Software development is configured into two or more sites [11] [9].</td>
</tr>
</tbody>
</table>

In the Categorization column, we present categories for FTS requirements. For each category, we map one or more characteristics of the FTS development scenario. These characteristics, we called requirements that satisfy an FTS scenario. Thus, ours findings (in Table II) contributed to characterize the FTS scenario, as:

In a FTS scenario, teams are into two or more sites geographically distributed in different time zones. An overlap between sites is timed to ensure synchronous transfer of tasks and provides software development in 24 hours per day. At the end of a working day, the team sends it to the next production location. The exchange task between teams is called handoff. Only one location at any point of time has the product.
We also compare characteristics from FTS scenario with GSD scenario traditional. Five characteristics are specifics of the FTS development scenario: (1) presence of the overlap between sites, (2) 24 hours development per working day, (3) handoffs at the end of a working day, (4) dependency of handoffs and (5) at any point in time only one location has the product. Four characteristics from FTS are shares with GSD: (1) Teams are geographically distributed, (2) Working teams in different time zones, (3) software development is adapted to different phases of the software development life cycle, and (4) software development is configured into two or more sites.

Our findings show that FTS has practices already conducted in GSD and it could be used in the FTS implementation. Key aspects for FTS implementation are related to five specifics characteristics. It contributes to difference FTS from GSD traditional.

B. GSD Practices for FTS Process (RQ2)

We consider five specific requirements of the FTS development scenario to identify GSD practices. Table III summarizes these practices found. Each practice is described as bellow.

<table>
<thead>
<tr>
<th>TABLE III. PRACTICES FOUND FOR FTS PROCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Categorization</td>
</tr>
<tr>
<td>Time zones</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Software development strategy</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Work collaborative</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

- **Time zones**
  1) *Communication tools*[29][30][31][32]

  We found that companies using communication tools in order to better structure their meetings [31]. According to Niinimaki et al. (2010), there are several communications media available for distributed software teams. Using of communication tools for FTS enables the communication between teams geographically distributed in different time zones. Wiredu (2011) emphasized the use of teleconference tools as positives for FTS.

  2) *Manage hand-on and shake-off sessions [31]*

Management of hand-on and shake-off sessions can help reduce communication and coordination problems between teams separated by different time zones. The overlapping sessions should be before the previous teams leave work, giving time for a team to hand-on their task to the next team and shake-off for the day. Managing these sessions effectively is significant because it helps to provide 24/7 support all round the year [31]. Three sub practices indicated by Deshpande and Richardson (2009) could be applied in handoff processes.

  1) There should be at least one hour overlapping session between two teams in different time zone.
  2) Discussion about what is done by the previous team and what needs to carried on by the next team should occur.
  3) Clear agenda for these sessions should be defined.

- **Software development strategy**

  3) *Establishing ‘Backup teams’ [31]*

  Establishing ‘Backup teams’ can help organizations to give support 24/7 all round the year and avoid any complex situations where, in particular, religious or national holidays do not coincide with western locations [31]. For this practice, Deshpande and Richardson (2009) indicate that at least ten percent of team members should be available to establish backup teams and such teams should be established both at national and international level.

  4) *Implement Tracking Systems [31]*

  A tracking system is a practice used in GSD to trace daily functioning as well as the overall performance of the teams and team members both at distributed locations. In FTS, this practice helps to manage day-to-day activities of the teams efficiently. It contributes to plan and keep check on any events that may cause project over-runs.

- **Work collaborative**

  5) *Documentation [30]*

  Documenting is well-known as not being the favorite task of software developers [33]. However, in collaborative development, the documentation is a way to capture details of the set up and configuration process and dealing with the complexity of the situation [30]. Setting this practice for FTS, we considered it necessary in situations for knowledge externalization and transfer.

  6) *Communication protocol [31]*

  Indian companies using a vertical and horizontal communication based in protocols. According to Deshpande and Richardson (2009), the model established by Indian companies, which mentions communication protocol among all team members, is simple, clear, appropriate, relevant, credible and non-overlapping communication. A communication protocol is a pre-defined model that defines horizontal and vertical channels of communication amongst team members and teams either collocated or located at the various geographies [31]. For FTS, communication protocols could reduce ambiguity and provide clear information in the handoff process. However, communication technologies used in
communication protocols as enablers of distributed work but they do not guarantee ‘location transparency’ or work FTS [4].

7) Making time zone differences manageable

In twenty-four hours scenarios, companies such as Intel, try to keep flexible and adjust hours to get a good overlap. The solution given by Intel is to make time zone differences manageable by dividing work between a limited number of the sites [4].

8) Creating time windows for interaction

To minimize inconvenience and conflicts in the collaboration between sites, a strategy is provided to create more opportunities for synchronous interaction. Shifting the workday, time windows for interaction reduce communication delays that could otherwise add a whole day or more in coordinating work between global sites [34]. This practice retained natural opportunities for communication without requiring advance planning and scheduling.

9) Use of collaborative technologies and knowledge sharing [35]

According to Gupta et al. (2009), using collaborative technologies and knowledge sharing achieve round-the-clock operation for the entire team and makes geographically distributed teams more effective.

V. ANALYSIS AND DISCUSSION

Analyzing results obtained, we found in eight hundred and nine papers only nine practices for FTS. Unfortunately, this result is not surprise to us. The low adoption of the FTS strategy in the software industry, made with the practices has not evolved. It makes sense, because there are a lack in practices and process to support specific needs of the FTS strategy [6].

In each practice described in this study, we found in the literature for its recommendation for FTS. We observe that these practices contribute for specifics FTS requirements. Although not described in the literature, evidence makes us believe that FTS is conducted in the software industry, but in part. According to Holmstrom et al. (2006), FTS concept is seen as one alternative to manage problems related to temporal distance. In 2006, HP Company related the use of FTS during Monday to Friday [4]. FTS practices support GSD needs, which are increasing in the software industry. According to Gupta et al. (2009), several organizations are seeking to develop software processes on twenty-four hour per day, with geographically distributed teams.

The characterization of the FTS scenario, we obtained by analysis of many studies, which discuss the software development twenty-four hours. We found definitions that give us information about configuration of the FTS scenario. The characterization of the FTS scenario proposed in this study, listed nine requirements, in which five are specifics of the FTS: (1) presence of the overlap between sites; (2) twenty-four hours development per working day; (3) handoffs at the end of working day; (4) dependency of handoffs; and (5) any point of time only a local has the product. These are the main requirements that make a difference between a traditional GSD scenario and FTS scenario.

Considering our findings, in particular the practices identified, we observed that effective use of the FTS strategy could be used in the software industry. The application of these practices to support needs of the development in twenty-four hours could encourage the development of new practices and processes for FTS.

In addition, we highlight that same practices found can be collaborate one with each other, such as, communication tools (1) and communication protocols (6) practices. In other words, it makes it possible to improve practices already conducted in the software industry.

With our findings, we observed that identified practices support both requirements for FTS as a traditional GSD environment. From the industry perspective, adopting of these practices can benefit creating opportunities to innovate in process and to support increasing needs of the software industry. Practices found give support on requirements for FTS implementation in time zones, software development strategy and collaborative work categories. From academic perspective, we highlight research opportunities in FTS encouraging studies about practices and processes.

VI. CONCLUSIONS

This paper presented results from a systematic mapping study (SMS) to identify GSD practices in the literature that could be used to support the FTS process. First, we collected information from the literature about FTS. With it, we categorized and pointed requirements of the FTS development scenario. We identified five specifics requirements. We also found nine practices for five specifics requirements of the FTS development scenario.

These results indicated important conclusions. Most requirements found in a development scenario indicated key aspects for FTS implementation, which are being implemented in the software industry in part. Practices found support FTS and GSD process partiality. Thus, we encourage the development of more researches about this.

Eight hundred and nine papers from academic and industry area were investigated, in which we analyzed solutions, learned lessons and practices given for GSD. We limited our study to identify practices conducted in GSD and recommended for FTS process. Thus, considering our findings, to continue the work, possible future studies should be carried out to identify associated practices with requirements FTS.

We highlighted in this study the characterization proposed for an FTS scenario, in which we were able to guide new studies and encourage the development of new practices to effective the use of the FTS strategy in the software industry.

ACKNOWLEDGMENT

The authors are funded by the PDTI program, financed by Dell Computers of Brazil Ltd. (Law 8.248/91).
REFERENCES


