



## Does follow-the-sun approach work for programming tasks?

S R Phanindra Kumar Narayanasetty<sup>1</sup> and Srikar Madhira<sup>2</sup>

1. M.Sc (S.E), BTH, Sweden

2. M.Sc (S.E), BTH, Sweden

**Abstract:** The main objective of writing this paper is to discuss the importance of global software engineering and listing the factors affecting the 'Follow the sun approach in solving Programming Tasks'. The various possible parties to overcome the effects that are encountered by following the 'FTS' approach are presented along with one possible alternative solution .

**Keywords:** *Global Software Engineering, FTS, Follow the Sun approach, Programming Tasks, alternative solution.*

### 1. Introduction

These days Global Software Engineering (GSE) has become a dominant paradigm in software engineering [1]. With the emergence of internet, the development of software has transitioned from traditional centralized, co-located form of development to a form in which software teams working for the same project or a system are collaborated across different geographical locations [1]. The benefits that motivate companies to promote Global Software Development (GSD) are reduction in time to time market, reduction in development costs, better use of scarce resources and business advantages from proximity to customers [1]. In the field of global software engineering (GSE), a concept exists by the name Follow The Sun (FTS) [2]. According to [3] FTS can be defined as "A type of global knowledge work flow, de-signed in order to reduce project duration, in which the knowledge product is owned and evolved by a production site and is handed-off daily to the next production site that is many time-zones apart to continue that work." [4]. The methodology involved in FTS is that by distributing a project

across multiple time zones, i.e. progress of working on project continuous round the clock by shifting it from one site to other site which are located geographically at a distant place.

### 2. Follow the sun approach

This section explains the concept of follow the sun approach which is an embedded concept of Global software Engineering. Following the sun approach may be an interesting approach transferring work from one location to another location many time zones away, in order to complete the project at a faster pace. Though the concept of FTS has huge documental success it is very difficult for implementation [2]. Consider a project distributed across two sites situated at two different geographical locations spread across different time zones. At the end of day 1 with respect to the local time at site-1, the work from site-1 is transferred to site-2. The further work is carried in site-2. Now after working on the task till day end at local time at site-2. The revised work is transferred to site-1 which started its workday and continues working

on the project from that point. The Figure 1 shows the above explained concept.

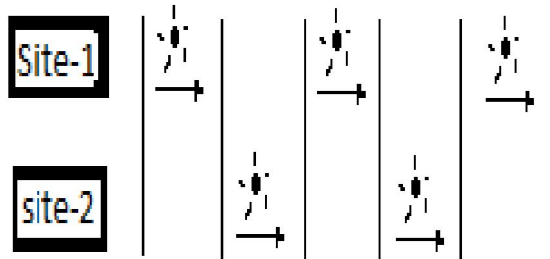


Figure1 showing the FTS Approach

Although FTS is an appealing approach, FTS has a low success rate which means that this approach is practically difficult to implement. This difficulty has not changed noticeably though there is much technical advancement in recent years. Hence currently this approach is not being used by organizations.

### 3. Implementing FTS Approach in Programming Tasks

Consider the example of solving an mathematical problem. There might exists a number of techniques to solve the problem. The perception of solving a problem differs from person to person. But the ultimate goal is to draw a conclusion to the problem. Consider implementation of FTS approach in solving programming tasks. An instance of implementing the FTS is shown in below figure.

From the figure 2, a programmer working at site-1 will work on a particular task in his own perception. At the end of the day the work will be transferred to another site which is situated at a different geographical location. The programmer at site-2 is now responsible for continuing the task of the same project from site-1.

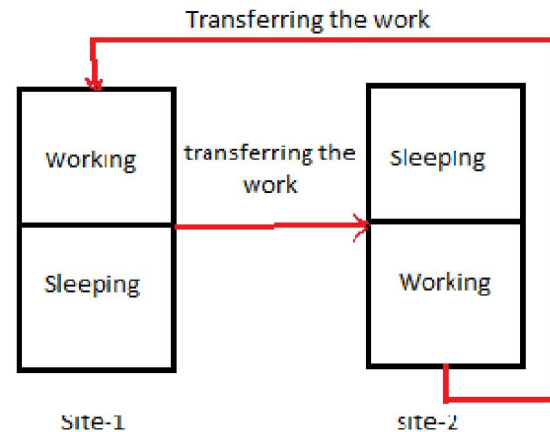


Figure 2 showing the transfer of work

The programme has two issues to deal

- The programmer has to understand what the previous programmer has done who is at a different geographical location.
- Secondly the programmer has to work on the remaining task and before the end of the day they should transfer the work done by them to another site.

The factors which effecting these issues of the programmer are given below [5]:

- **Lack of Communication between the two programmers.**

The communications between the two programmers is asynchronous and done only via E-mails. If the programmer at site-2 faces any problem regarding the task done by programmer at site-1 then due to lack of face to face communication between the programmers the development of task takes more than normal time to complete and the programmer at site-2 must sit ideally until the reply from other site programmer[9]. The lack of formal communication [6] decreases trust level.

- **Culture difference**

There were number of issues which were caused by unknown differences in social morals and cultural values. For instance if there exist any problem in understanding the work done at site-1 the below are the reasons that sit 2 programmer may face due to culture impact. Firstly one culture tends to have more emphasis on self sufficiency hence they feel that they are competent enough to solve their own problem. Another case is like that they would not offer help unless they are asked for help[8].

- **Technical environment**

Consider the programmer from site-1 has knowledge on programming and has vast experience. The programmer from site-2 has no experience in programming. Then he focuses on getting the work done by learning new tools and techniques without any experience in programming background. Also the programming practises vary from one location to another location.

We can reduce the above factors by following some of the practises which are shown below [5].

**Boot Camps:** To overcome the lack of trust between the developers of both the sites. Boot camps are organized. This is done to build up the trust level between the developers.

**Round-the-world program:** Each and every country differs from one another. Traditionally and cultural software organization makes decisions according to specific local conditions. If there is vast difference between the region then the ideas implemented also differs .If this continues there is possibility of developing misunderstanding. To mitigate this problem of misunderstanding and to build up trust, round the world technique has been introduced. According to this approach, one member from a region shift to work in another

region for some time this process of shifting continuous eventually till the members of the team gain experience of working in all the regions. Trust can be built when relationships are built over long periods of time. But, due to family or funding problems this may not me suggestible.

**Routing guru:** Unfamiliarity of different tools and techniques in one region delays the development. To mitigate this challenge an experienced team member who is more familiarly with the development tool and techniques is sent to the other region for taring the developers at the other site. This help to develop the task as soon as possible.

#### 4. Alternative Solution

Communication, cultural difference and technical background are major problems in FTS as discussed in previous section. We may reduce the communication and cultural difference by organizing the development in one site around the clock in similar fashion of FTS. Some countries like India have ample man power and willing to work at any time during the day. We can organize follow the clock approach for development the task so, that we can reduces the communication and cultural problems beside we can also reduces the problems related to handling the work to other site the development structure is as shows from figure 3.

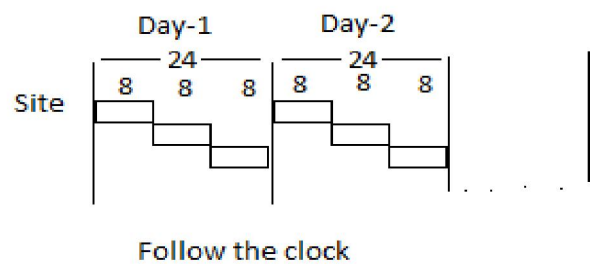


Figure 3 follow the clock

The figure shows 3 development sites in one site as pre clock duration. At first one team will work on the task at the end of the clock time of 8 hours then handover work to second team after that completing their work it was transferred to team three and final team three will return the work to team one. By this way we can reduce the problems related to communication, transfer and trust. Where the expected outcomes of this approach is similar to FTS.

## 5. Conclusion

In this paper, we discussed the factors affecting when we follow the FTS approach for the programming task and some of the practices to minimize it. Mainly FTS is used to reduce the development time and its cost [1]. The development time can be reduced by 50% when two sites and if three sites 67% [2]. But when coming to programming task it was quite complicated because there are many ways to solve the task it will depend on individual that to with asynchronous communication with other person besides with problems with culture difference and technical environment. Some of the studies insist that FTS is not suitable for developing rather than other phases like testing, maintenance and documentation [6,7]. But a well structure FTS and by using some of agile methods we can overcome these issues [5,10]. FTS is not suitable with traditional global software development rather it is well suited with agile development when we use some of its core practices of agile [11].

## References

Carmel, E.; Dubinsky, Y.; Espinosa, A.; , "Follow The Sun Software Development: New Perspectives, Conceptual Foundation, and Exploratory Field Study," *System Sciences, 2009. HICSS '09. 42nd*

*Hawaii International Conference on* , vol., no., pp.1-9, 5-8 Jan. 2009.

E. Carmel, Y. Dubinsky, and J. Johnston. Follow the sun work flow in global software development: Theory, modeling and quasi-experiment to explore its feasibility. In *Proceedings of the Third Global Sourcing Workshop: The Impacts of Global IS Sourcing on Engineering, Technology and Innovation Management*, Key-stone, CO, USA, March 2009.

Erran Carmel, J. Alberto Espinosa, Yael Dubinsky, "Follow the Sun" Workflow in Global Software Development", *Journal of Management Information Systems*, volume 27,17-38,2010.

Global software development: technical, organizational, and social challenges Filippo Lanubile, Daniela Damian, Heather L. Oppenheimer November 2003 SIGSOFT Software Engineering Notes , Volume 28 Issue 6.

Helen, Milla; Nahar, Nazmun; , "Key barriers of globally distributed software products development," *Technology Management in the Energy Smart World (PICMET), 2011 Proceedings of PICMET '11*: , vol., no., pp.1-13, July 31 2011-Aug. 4 2011.

Helena Holmstrom, Eoin O Conchuir, Par J Agerfalk, Brian Fitzgerald, "Global Software Development Challenges: A Case Study on Temporal, Geographical and Socio-Cultural Distance," *Global Software Engineering, 2006. ICGSE '06. International Conference on* , vol., no., pp.3-11, Oct. 2006.

J. J. Treinen, S. L. Miller-Frost," Following the sun:Case studies in globalsoftware development IBM SYSTEMS JOURNAL, VOL 45, NO 4, 2006.

Linna, P.; Karttunen, E.; Jaakkola, H.; , "Software engineering companies' multicultural education," *MIPRO, 2011 Proceedings of the 34th International Convention* , vol., no., pp.1140-1145, 23-27 May 2011.

Setamanit, S.-o., Wakeland, W., Raffo, D.; , "Improving Global Software Development Project Performance Using Simulation," *Management of Engineering and Technology, Portland International Center for* , vol., no., pp.2458-2466, 5-9 Aug. 2007

Visser, C.; van Solingen, R.; , "Selecting Locations for Follow-the-Sun Software Development: Towards a Routing Model," *Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on* , vol., no., pp.185-194, 13-16 July 2009.

Yap, M.; , "Follow the sun: distributed extreme programming development," *Agile Conference, 2005. Proceedings* , vol., no., pp. 218- 224, 24-29 July 2005