

A Structured Approach to Global Software Development

*Valentine Casey
Bournemouth University
Poole
Dorset
United Kingdom*

*& Ita Richardson
University of Limerick
Ireland*

Abstract

The analysis of the combined results from three independent industry focused case studies, undertaken in the area of distributed software development over a period of eight years, has resulted in the identification of ten key factors. These ten factors have been utilised as the basis for the development of the *GSD Implementation Model*. The objective of the creation and presentation of this model is to provide a practical and systematic approach to address the key activities, infrastructure and support which are required to facilitate effective distributed software development. This approach is inspired by the IDEAL model and divided into five specific phases which are classified as Initiating, Provisioning, Establishing, Managing and Leveraging. The goal of the Initiating phase is to clearly determine why, if and how the distributed development strategy is to be selected and undertaken. The implementation of the Provisioning phase is to ensure that the required infrastructure, processes and support to facilitate successful distributed software development are identified and put in place. The focus of the Establishing phase is to ensure that the development teams are effectively established. The managing phase addresses the day to day requirements of operating efficiently in a distributed environment. The Leveraging phase concentrates on the need to ensure that the structures and procedures are in place so that lessons learned can be documented and leveraged in existing and future projects.

Keywords

Global Software Development, GSD, Virtual Teams, Outsourcing, Offshoring, Infrastructure, Software Process, Risk, Project Management, Culture, Communication, Co-ordination.

1 Introduction

In today's highly integrated international markets software development is considered a globally sourced commodity [1]. The sustained popularity for the selection of this strategy is ascribed to organisations endeavouring to gain and maintain competitive advantage from the globalization of software development [2]. The potential for achieving this advantage is attributed to the benefits provided by labour arbitrage, which offers the opportunity for re-

duced development costs [3]. This continues to be facilitated by the availability of well educated and technically competent software engineers in low cost centres in Eastern Europe, Latin America, India and the Far East [4, 5]. It is a commonly held belief that these savings can be coupled with the opportunity for round the clock development facilitated by the temporal difference between remote development locations. The logic underpinning this approach is that these two factors can facilitate competitive pricing and reduce time to market. Thus enabling companies to compete more effectively by gaining, expanding or maintaining their market share [6].

As many organisations who have implemented a Global Software Development (GSD) strategy have discovered, due to the level of complexity involved in software development, outsourcing to other organisations or offshoring to remote divisions is not a straightforward task [3, 6-8]. Some of the difficulties encountered include such factors as the problem of understanding requirements, testing of systems and the coordination of these types of projects [7]. These difficulties are further compounded by cultural and language differences, lack of communication, geographical and temporal distance from team members and the customer, different process maturity levels, development and testing tools, standards, technical ability and experience. As a result the management of globally distributed software development projects has been recognised as a difficult and complex task [9].

Given all these circumstances it is not surprising that offshoring and outsourcing software development has proved a complex endeavour and should never be embarked on lightly or without due consideration. A major problem which has emerged in this area is that too often the implementation of an outsourcing or offshoring strategy has been seen as simply the replication of those strategies which are implemented for collocated software development. This short sighted approach has led to serious problems and numerous failures [2, 7]. It is in this context and with the objective of helping to address the issues which have been outlined the authors have undertaken to develop the *GSD Implementation Model*.

2 Three Independent Case Studies

The findings presented in this paper are based on the results from three independent case studies which the authors have undertaken over an eight year period in the area of distributed software development. The first case study was carried out in an Irish company called Irish Computing Solutions (a pseudonym) who implemented a strategy to expand their organisation's market share by the establishment of local offsite virtual software development teams. Prior to implementing this policy the company operated collocated teams based in the capital (Dublin) who worked exclusively on the development of financial and telecommunications software. In addition the organisation had a software development centre located 150 miles from Dublin. This centre was involved in general application development and maintenance and had lower labour costs than the capital. The objective was to leverage staff at both locations and capitalize on the cost advantage which this strategy offered. A group of twelve offsite engineers were selected and were provided with basic training in the technology and process required. Two virtual teams were established and consisted of two sets of six offsite engineers who were partnered with three experienced onsite engineers based in Dublin. Considerable effort was put into providing the communication infrastructure, process and support for both virtual teams. A key objective of this approach was that the onsite engineers would mentor the inexperienced offsite staff and provide effective knowledge transfer. The operation of these teams and their subsequent failure provided the basis for this case study [10].

The second case study focused on what is termed offshore / nearshore software development [1]. The concept of offshore / nearshore is derived from the fact that the research centred on a partnership between a large US based financial organisation Stock Exchange Trading Inc. and an Irish division of a US multinational company Software Future Technologies (both pseudonyms). The US and Irish based sites were geographically distant, but they were considered linguistically and culturally nearshore [1, 11]. This partnership ultimately resulted in the establishment of virtual teams to develop and maintain bespoke financial software. Stock Exchange Trading Inc. was the senior partner in this relationship and had an on going requirement for the development and maintenance of this type of software. An unanticipated and urgent requirement arose for the development of new software during the initial stage of establishing the virtual teams. To address this need 70 percent of the Irish team members moved to the US, as a temporary measure for a period of one year to work on collocated teams with their Stock Exchange Trading colleagues. This proved to be a very effective strategy and both groups operated very successfully while collocated within what were to eventually become their virtual teams. It was only when the Irish team members returned to Ireland and the virtual teams were established that serious problems arose. These problems and issues and their ultimate solution have been articulated in detail in [10, 12, 13].

The third case study centred on offshore virtual team software testing and was undertaken in the Irish division of a large US multinational called Computing World International (a pseudonym) who had been operating in Ireland for over twenty years. The Irish division had been very successful and had expanded considerably over that time. During that period a large percentage of the projects undertaken had been offshored from their US parent; therefore, the Irish staff and management were very experienced in having projects offshored to them

Two years prior to undertaking this case study the organisation's corporate strategy changed. At that time they initiated a policy of establishing virtual testing teams with the objective of leveraging the technical ability of their Irish staff with the competitive salary levels of their Malaysian test engineers. When this research commenced four virtual testing teams were in operation between the Irish and Malaysian divisions. Some teams were established for over a year and a half while others had only been in operation for a number of months.

This case study focused on two embedded units of analysis. One was a virtual testing team with members located in Ireland and Malaysia which had been in operation for a period of eighteen months. The second was a virtual team with a similar makeup, but had been established for just over six months. The different aspects and findings from this study have been outlined in detail and published in [10, 13-15].

2.1 Research Methodologies

The research methodology employed in the first and second case studies was the action research five-phase cyclical process based approach as defined by Susman and Evered [16] and Baskerville [17]. Action research entails the analysis of the direct intervention of the researcher. This methodology was selected as the most appropriate for both case studies as one of the authors held a management role in the respective organisations researched. The objective in both situations was to leverage the research opportunities which this provided while maintaining the required level of objectivity of both researchers. The third case study required a different approach and research methodology. When this study was undertaken both authors were fulltime researchers and were offered the opportunity to undertake extensive on

site research. The objective was therefore to maximize the level of access this opportunity provided. After due consideration this resulted in the selection and implementation of a Yin [18] based embedded case study which incorporated a Strauss and Corbin grounded theory [19] approach to data gathering and analysis.

3 The Development of the GSD Implementation Model

Based on the analysis of the combined results from the three case studies [10, 12-15] ten key factors were identified. It was determined these factors were directly relevant and needed to be specifically addressed in order to establish and facilitate the operation of globally distributed virtual teams. These factors are summarised as follows:

1. Understand why, at what cost and risk a distributed strategy is undertaken
2. The Provision of effective infrastructure, process and documentation
3. The requirement to effectively establish the teams
4. Implement an efficient distributed team project management strategy
5. Ensure the development of common goals, objectives and rewards
6. The need for the clear definition of roles and responsibilities
7. Address issues related to culture, communication, motivation and fear
8. Ensure provision of adequate training and knowledge transfer
9. Facilitate and monitor the operation of collaborative and supportive teams
10. Document and leverage lessons learned

3.1 Foundation of the Model

Reviewing the ten key factors which were identified by this research it was determined of value to consider how they could be utilised to develop a strategy for the establishment, operation and the effective management of virtual software teams. It was realised they also had relevance and implications for GSD in general. To address both of these issues a model was developed which highlighted the key areas which needed to be considered and addressed to facilitate successful virtual team operation and globally distributed software development.

When developing this model it was recognised that it required to be clear so that it could be easily understood and implemented, to be practical so that it would be used and to be comprehensive to address the numerous relevant factors and issues which impact on GSD. It was also required to incorporate an element which facilitated recording relevant experience and knowledge gained while establishing and operating the GSD teams. This could then be leveraged to improve existing operations and assist with the implementation of GSD strategies in the future.

It was in this context that the IDEALsm model [20] was researched and identified as an appropriate basis for the development of the *GSD Implementation Model*. The original focus and application of the IDEALsm model is in the area of Software Process Improvement (SPI). In these circumstances the authors had in previous research utilised it as an effective tool and its adaptability had been successfully implemented to achieve SPI [21]. Its wider applicability and potential for use outside this specific SPI area has been recognised by the Software Engineering Institute (SEI). It is acknowledged that the model can provide an effective and disciplined approach for the adoption of new software engineering processes, methods and tools. In these circumstances it can also be utilised for establishing the foundation for and the maintenance of a long-term improvement strategy [22].

It was recognised that the IDEALsm model presented a structure which could be amended to directly address all the relevant requirements and areas of concern which impact on the establishment and operation of GSD teams. It provided a simple, but comprehensive framework on which the *GSD Implementation Model* could be based. It also offered a straightforward, practical and extensive approach. Based on all these factors it was considered suitable. It has been adapted to the specific requirements of the GSD environment. What was proposed was not to mirror the IDEALsm model in every aspect, but to utilise its relevant constituent parts and overall approach. Therefore the development of the GSD Implementation Model was based on the basic structure of the IDEALsm model which was expanded and modified to meet the specific requirements and needs of operating in the globally distributed software development environment.

4 The GSD Implementation Model

The ten key factors which our research identified were divided into five distinct phases, which were to be undertaken sequentially. The model as a whole was designed for iterative execution (see figure 1). The five phases are as follows:

Initiating – Determine why, if and how the GSD approach is to be implemented

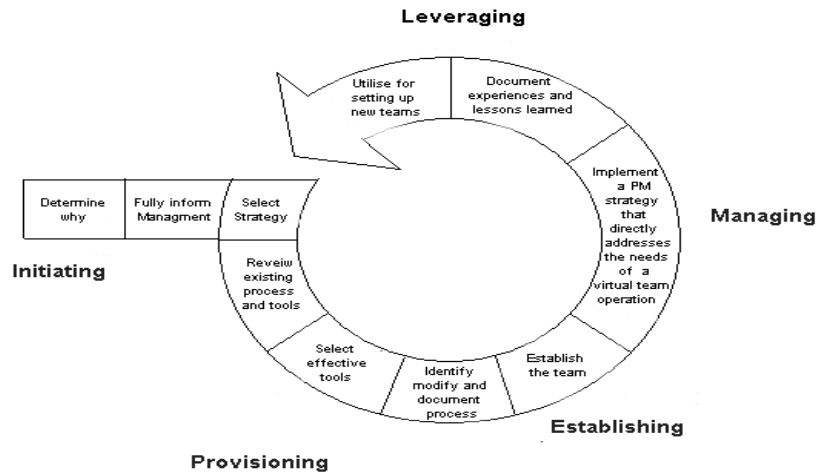
Provisioning – Ensure provision of effective infrastructure, process and documentation

Establishing – The requirement to effectively establish the GSD teams

Managing – Implementation of an efficient GSD project management strategy

Leveraging – Document and leverage lessons learned for existing and future projects

sm IDEAL is a service mark of Carnegie Mellon University



The GSD Implementation Model
Figure 1

4.1 Initiating

There is a requirement for organisations considering outsourcing or offshoring part or all of their software development activities to clearly define and articulate their rationale for selecting and implementing such an approach. In some cases justification is simply the result of a perceived cost advantage of implementing a GSD strategy at a corporate level or the fact that competitors are doing it. In a number of situations this type of rationale has proved very short sighted and led to serious problems. In these circumstances it is important that organisations recognise that the reality can be quite different. GSD projects can and have ended up costing as much or more than if they were collocated. They can also negatively impact on the delivery and quality of the software artefacts produced and the morale and motivation of existing staff [10, 13, 15].

In addition risk is a key factor which needs to be specifically addressed in the GSD environment, while pervasive risk should be incorporated into all well planned software projects [23, 24]. Globally distributed development projects carry additional high risk exposure [25]. These include the risk of delay or failure due to linguistic, cultural difference, motivation and temporal distance. All these issues need to be recognised and understood prior to embarking on or implementing such an approach [2, 26]. This can only take place when time is spent gathering and evaluating information on exactly what is involved and what are the positive and negative factors which are inherent to operating in a GSD environment.

If it is decided this is the strategy the organisation wishes to implement, the real potential costs and risks involved need to be accurately assessed. Based on these realistic projections the objectives of the strategy should be determined and directly linked to the short and long-term goals of the organisation. Senior management support is key to the success of any distributed software development strategy. Therefore, they must be provided with all the information necessary to allow them to have realistic expectations as to what can be actually achieved. Once the decision to implement this approach has been agreed the most appropriate GSD strategy should be selected.

4.2 Provisioning

Having selected a GSD strategy the infrastructure to support its implementation needs to be determined and put in place. In this context existing tools and processes need to be reviewed, adjusted and augmented. In some low cost locations the availability of a dependable electrical supply and alternative power source need to be considered and addressed. Of equal importance is the availability of an adequate telecommunications infrastructure. Once basic infrastructure has been established across the relevant sites common or compatible tools need to be identified and sourced. This is required to ensure the interoperability of cross-site operations and artefacts. In this context an essential aspect of GSD is the selection and implementation of an effective configuration management system [7]. Due consideration also needs to be given to the selection of appropriate communication tools which are essential when operating in what can largely be an asynchronous environment [3, 12, 14].

Once adequate infrastructure is in place the identification and adoption of a common and effective GSD process needs to be considered [7]. Organisations must reassess and modify their existing processes for use in a distributed environment [26]. This includes the need for more formal methods of collaboration and communication given the loss of informal communication methods [27]. In the GSD situation there is a clear need for a well-defined jointly formulated and documented process to be put in place [21].

4.3 Establishing

The next step is to effectively establish the teams. Team members should be recruited internally and externally based on the technical needs of the project. Provision should be made for technical, cultural and communications training which are specific to the needs of the GSD environment [15]. The foundation for effective knowledge transfer between team members regardless of location should be put in place. This includes leveraging all visits between team sites to develop relationships. A priority from an initial stage is the establishment of a one-team vision and cooperative approach between team members regardless of location. This has to be actively fostered, developed and monitored[14].

4.4 Managing

There is the need for the development and implementation of an efficient GSD project management strategy which incorporates and addresses the specific requirements of operating in a distributed environment [14, 15]. In this context there is a need to facilitate and ensure the development of common goals, objectives and rewards. This is achieved by specifically addressing the issues, factors and variables that GSD teams are exposed to [3, 7]. There is also the requirement for roles and responsibilities to be clearly defined and articulated to all managers and team members. This is achieved through the use of a common vocabulary which unambiguously outlines this information.

There is also a requirement to address issues which are specifically related to culture, communication, motivation and fear [10]. This is achieved by understanding these issues and ensuring they are monitored and that timely and corrective action is taken to address any problems which arise due to any of these areas. Of equal importance is to monitor the effectiveness of technical training and knowledge transfer. When the requirement for additional

training is identified it should be provided. If problems are identified with knowledge transfer they need to be investigated and specifically addressed. There should also be incentives to encourage staff to effectively transfer knowledge.

A cohesive team does not emerge of its own accord from a globally distributed, culturally, linguistically and technically diverse group of individuals, who are separated by geographical and temporal distance [7]. If it is to be put in place, it requires effort and goodwill on all sides. It can happen, but it must be planned, established, supported, monitored and actively developed. It can only take place with effective management where the positive aspects of the GSD environment are effectively leveraged and the negative factors and issues are addressed [12].

4.5 Leveraging

A key activity is leveraging the experience and knowledge gained by implementing a GSD strategy. This is best achieved by analysing and documenting the experience and knowledge gained. This should then be utilised to review what has been achieved and identify areas where further improvements can be made. This information should also be made available and used to directly assist with the management of other existing teams and the establishment and operation of new GSD projects.

5 Conclusion

The *GSD Implementation Model* provides an overview which is practical and comprehensive in its structured and iterative approach. Within its five phases it addresses the specific requirements of operating in a GSD environment. This is achieved by ensuring the rationale for undertaking this approach is clearly articulated and understood and that realistic objectives and goals are set. Senior management support is secured on achievable expectations based on the accurate evaluation of costs and risks. The required infrastructure, processes and supports are put in place to facilitate the operation of the GSD teams. Time and effort is put into effectively establishing and managing the teams. An effective project management strategy based on the needs of the GSD environment is implemented. Key to the long term success of this approach is the documenting and leveraging of the experience gained implementing such a strategy. This model has been presented to forty five senior managers who had direct experience of implementing GSD strategies for evaluation. Their response was very positive and the consensus was that it was an excellent model to utilise when embarking on a GSD strategy as it highlighted the key areas which need to be specifically addressed.

6 Literature

1. Hayes, I.S., Ready or Not: Global Sourcing Is in Your IT Future. Cutter IT Journal, 2002. 15(11): p. 5 - 11.
2. Prikladnicki, R., J.L.N. Audy, and R. Evaristo, Global software development in practice lessons learned Software Process Improvement and Practice, 2003. 8(4): p. 267 - 279.
3. Karolak, D.W., Global Software Development: Managing Virtual Teams and Environments. 1999, Los Alamitos, CA, USA IEEE Computer Society Press.

4. Carmel, E. and R. Agarwal, Tactical Approaches for Alleviating Distance in Global Software Development. *IEEE Software*, 2001. 1(2): p. 22 - 29.
5. Crow, G. and B. Muthuswamy, International Outsourcing in the Information Technology Industry: Trends and Implications. *Communications of the International Information Management Association*, 2003 3(1): p. 25 - 34.
6. Herbsleb, J.D. and D. Moitra, Global Software Development. *IEEE Software*, 2001. 18(2): p. 16 - 20.
7. Carmel, E., *Global Software Teams: Collaboration Across Borders and Time Zones*. 1999, Saddle River, NJ: Prentice Hall.
8. Clerc, V., P. Lago, and H. van Vliet. Global Software Development: Are Architectural Rules the Answer? . in *Second IEEE International Conference on Global Software Engineering*, 2007. ICGSE 2007. 2007. Munich: IEEE.
9. Lanubile, F., D. Damian, and H.L. Oppenheimer, Global software development: technical, organizational, and social challenges *SIGSOFT Software Engineering Notes*, 2003. 28(6): p. 1 - 4.
10. Casey, V. and I. Richardson, The Impact of Fear on the Operation of Virtual Teams, in *International Conference on Global Software Engineering*, ICGSE 2008, IEEE: Bangalore, India.
11. Sahay, S., B. Nicholson, and S. Krishna, *Global IT Outsourcing: Software Development across Borders*. 2003, Cambridge UK: Cambridge University Press.
12. Casey, V. and I. Richardson. Practical Experience of Virtual Team Software Development. in *Euro SPI 2004 European Software Process Improvement*, . 2004. Trondheim, Norway.
13. Casey, V. and I. Richardson. Virtual Software Teams: Overcoming the Obstacles. in *3rd World Congress for Software Quality*. 2005. Munich, Germany.
14. Casey, V. and I. Richardson. Uncovering the Reality within Virtual Software Teams. in *First International Workshop on Global Software Development for the Practitioner*, ICSE 2006 2006. Shanghai, China ACM Press
15. Casey, V. and I. Richardson. Project Management within Virtual Software Teams. in *International Conference on Global Software Engineering*, ICGSE 2006. 2006. Florianopolis, Brazil: IEEE.
16. Susman, G. and R. Evered, An Assessment of the Scientific Merits of Action Research, . *The Administrative Science Quarterly*, 1978. 23(4): p. 582 - 603.
17. Baskerville, R.L., Distinguishing Action Research from Participative Case Studies *Journal of Systems and Information Technology*, 1997. 1(1): p. 25 -45.
18. Yin, R.K., *Case study research / design and methods* Second ed. *Applied Social Research Methods* Vol. 5. 1994, Thousand Oaks, CA. USA: Sage Publications.
19. Strauss, A. and J. Corbin, *Basics of Qualitative Research : Techniques and Procedures for Developing Grounded Theory*. Second ed. 1998, Thousand Oaks, CA, USA: Sage Publications.
20. Paulk, M.C., et al., *The Capability Maturity Model: Guidelines for Improving the Software Process*. *SFI Series in Software Engineering*. 1997, Reading, MA. USA: Addison-Wesley.
21. Casey, V. and I. Richardson, A Practical Application of the IDEAL Model. *Software Process Improvement and Practice*, 2004. 9(3): p. 123 - 132.
22. Gremba, J. and C. Myers, *The IDEAL Model: A Practical Guide for Improvement*. Bridge, 1997(3).
23. van Vliet, H., *Software Engineering : Principles and Practice*. 2nd ed. 2000, Chichester, U.K. : Wiley.

24. Henry, J., *Software Project Management: A Real-World Guide to Success* 2003, Boston, MA, USA: Addison-Wesley.
25. Ebert, C., et al. Improving validation activities in a global software development in *Proceedings of the 23rd International Conference on Software Engineering 2001* Toronto, Ontario, Canada IEEE Computer Society.
26. Nidiffer, K.E. and D. Dolan, Evolving distributed project management. *IEEE Software* 2005. 22(5): p. 63 - 72
27. Herbsleb, J.D. and R.E. Grinter, Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software*, 1999. 16 (5): p. 63 - 70

NB This is a prepublication version of this paper

