

A Maturity Model for Offshore Insourcing: A Research Proposal¹

Jairo Avritchir

jairo_avritchir@uol.com.br

Roberto Evaristo

College of Business Administration
University of Illinois at Chicago
evaristo@uic.edu

Rafael Prikladnicki

Computer Science School
PUCRS
rafael@inf.pucrs.br

Jorge Luis Nicolas Audy

Computer Science School
PUCRS
audy@inf.pucrs.br

ABSTRACT

The objective of this paper is to present a research proposal toward a maturity model for distributed IT projects for those organizations involved with offshore insourcing. Although offshore insourcing bypasses some of the tough contracting difficulties found in organizations that are involved with traditional offshore outsourcing, a whole different set of issues is created. However, some of the “new” issues that surface a particular organization may have already been dealt with by more experienced organizations. The maturity model will therefore not only identify best practices along difference maturity stages but also help to benchmark against similar organizations. Our research suggest that there is a very consistent sequence of phases that this particular type of IT offshore sourcing follows, and the study is based on a rigorous research method. We expect to have preliminary results by August 2004.

Keywords

Offshore Sourcing, Offshore Outsourcing, Offshore Insourcing, Distributed Software Development, Maturity Model.

INTRODUCTION

Software has become a vital component of almost every business. Success increasingly depends on using software as a competitive advantage (Herbsleb, 2001). More than a decade ago, many organizations began to experiment with remotely located software development facilities (also called Distributed Software Development - DSD) seeking lower costs and access to skilled resources. Economic forces are relentlessly turning national markets into global markets and spawning new forms of competition and cooperation that reach across national boundaries (Carmel, 1999).

This change is having a profound impact not only on marketing and distribution but also on the way products are conceived, designed, constructed, tested, and delivered to customers. For these reasons, DSD has attracted a large research effort in software engineering (i.e., Karolak, 1998; Carmel, 1999; Herbsleb, 2001; Damian, 2002; Prikladnicki, 2003). The search for such competitive advantage forces organizations to search for external solutions in other countries (offshore sourcing). This epitomizes the traditional problems and the existing challenges.

The two main options currently under use include offshore outsourcing (contracting services with an external organization located in another country) as well as offshore insourcing (contracting with a wholly owned subsidiary also located in another country). The first has become fairly common, but difficulties abound in trying to develop a relationship with an unknown foreign partner that is time and geographically distant. Such issues have led select organizations to create their own software development centers in countries like India, Russia, Brazil, Ireland, etc. Although offshore insourcing bypasses some of the tough contracting difficulties found in organizations that are involved with traditional offshore outsourcing, a whole different set of issues is created. And one of the key concerns that typically surfaces is “how other organizations in similar situations are dealing with these problems?” The purpose of this research is to propose a maturity model for offshore insourcing.

THEORETICAL MODEL

Distributed Software Development

In the last decade there have been large investments to enable the move from local to global markets, in the process of creating new competition and collaboration forms (Herbsleb, 2001). The global software market, however, has undergone several crises: Not only a large number of project failures have plagued the industry, but also the increasing demand for new systems has been strongly affected by scarcity in appropriate competences. In such environment, DSD provides a feasible alternative. Several factors have accelerated this trend (Herbsleb, 2001):

- The need to have a global resource pool to successfully and cost-competitively have resources, wherever located;
- The business market proximity advantages, including knowledge of customers and local conditions;
- The quick formation of virtual corporations and virtual teams to exploit market opportunities;
- Pressure to improve time-to-market by using time zone differences in “round-the-clock” development;

Organizations search for competitive advantages in terms of cost, quality and flexibility in software development, looking for productivity increases as well as risk dilution (Prikladnicki, 2003). Many times the search for these competitive advantages forces organizations to search for external solutions in other countries (offshore sourcing). This epitomizes the traditional problems and the existing challenges as follows (Herbsleb, 2001):

- Strategic issues: the decision whether a particular project can be developed by globally dispersed teams – and where it can be better developed, as well as how to divide it across sites – is difficult. Some analysis considering the risk and benefit of projects dispersion can be necessary. Solutions are constrained by the resources available at the sites, their levels of expertise in various technologies, infrastructure, etc;
- Cultural issues: offshore sourcing requires close cooperation of individuals with different cultural backgrounds. Cultures differ on many critical dimensions, such as national, ethnic, organizational, professional, technical, and team culture. While many people find such differences enriching, they can also lead to serious and chronic misunderstanding, especially among people who do not know each other well;
- Technical issues: when teams are working across sites, the lack of synchronization can be particularly critical. We need to assure a common defined milestones and clear entry and exit criteria for all tasks. The overhead of control (the process of adhering to goals, policies or standards) and coordination (the act of integrating each task and organizational unit so that it contributes to the overall objective) associated with any software projects is astounding;
- Knowledge management: without effective information and knowledge-sharing mechanisms, it is difficult to exploit the benefits of offshore sourcing. Poor documentation can cause ineffective collaborative development.

According to Carmel (1999), global team formation requires the identification of centrifugal and centripetal forces. He sees software globalization as five centrifugal forces (coordination breakdown, loss of communication richness, geographic dispersion, cultural differences, and loss of “teammess”) that propels things outwards from the center. A centrifugal force must be balanced by a centripetal force (product architecture, telecomm infrastructure, team building, managerial techniques, collaborative technology, and development methodology), a counter force that is directed to the center.

Moreover, (Evaristo 2003) suggests dimensions to the concept of “distributedness” through a theory-based model. The main objective is to understand what “distributed” means when discussing the management of distributed projects and to suggest better ways to manage them by finding out what are the critical problems. These dimensions (Figure 1) are related not only to software development projects, but also to more general distributed projects.

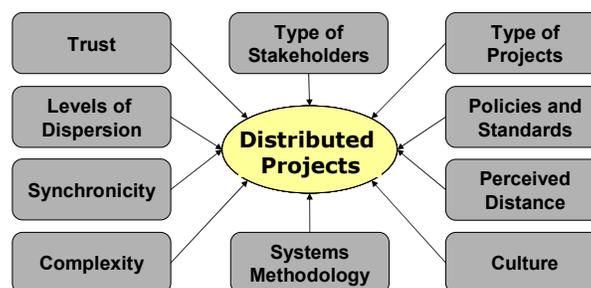


Figure 1. DSD dimensions, according to (Evaristo, 2003)

For the purpose of this research project, we highlight trust and cultural differences.

Trust: Trust is critical to keep the *esprit-de-corp*, or teamness, in a team; however, it is something difficult to create, keep and manage on the day-to-day operation of a DSD team. Distributed relationships are clearly encumbered by distance, and trust is arguably one of the most important ingredients for performance in a global team.

Culture: Culture is important even in co-located projects because of its pervasive impact on attitudes and team behaviours. It is easier to notice and clarify misunderstandings in co-located situations due to the presence of rich media clues like expressions or even seemingly unrelated comments. In the case of distributed projects, however, cultural differences that may pass unnoticed magnify any problems that may occur because of the inherent difficulty in distributed environments to notice and correct potential misunderstandings. Worst yet, such perception may only develop too late.

DSD dramatically impacts the way products are conceptualized, developed, tested and delivered. Therefore, the structure needed to support such environment is also completely different, both in terms of processes as well as technology. Details not before obvious grow in importance, and this research will uncover some of these issues as well.

Offshore Sourcing

Offshore sourcing of IT work is increasingly occupying the attention of IT managers in U.S.-based firms. The term “offshore sourcing” includes both offshore outsourcing to a third-party provider as well as offshore insourcing to an internal group within a global corporation (Carmel, 2002). Organizations that avail themselves of outsourcing services can concentrate in its core businesses, potentially reducing the software development team. The combination of these factors results in a significant reduction in time and cost of software development.

IT managers are being pressured to contain costs in addition to ramping up projects quickly, finding experienced staff in fast-moving technologies, and innovating constantly with IT. To acquire the IT competencies that address these challenges, managers can choose one of two strategies: either outsource to a domestic supplier or go offshore. The foreign sourcing of IT work is growing based mainly on:

- The increasingly modular design has reduced transactions costs – that is, the cost of coordinating software development and support work between two or more parties. More modular software production eases the burden of synchronizing, communicating, traveling, monitoring, providing feedback, and enforcing contracts;
- Technologies for managing and coordinating work across geographic distances have matured considerably;
- Offshore organizations have improved their software development and project management capabilities.

In the study conducted by Carmel (2002), offshore IT sourcing was identified as a maturation process. The authors proposed the following four stages in a model called SITO: offshore bystander, offshore experimenter, proactive cost focus, and proactive strategic focus (Figure 2).

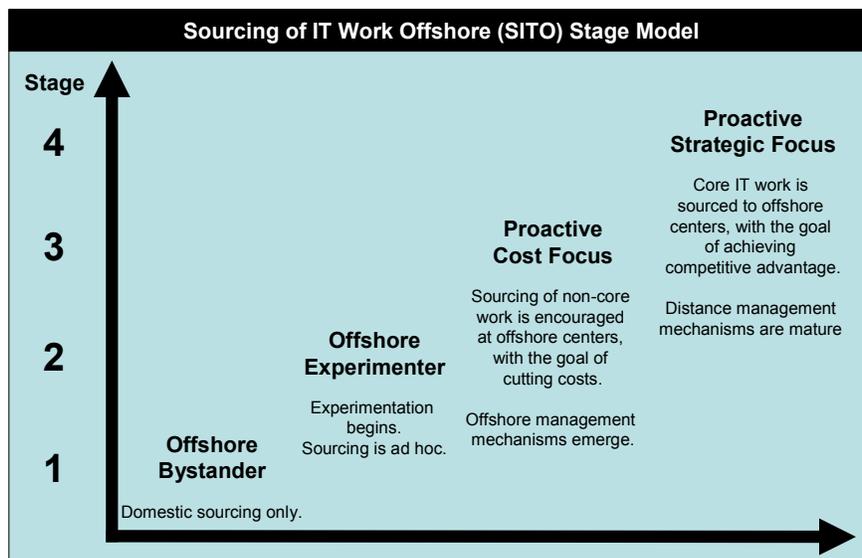


Figure 2. The SITO Model (Carmel, 2002)

Each stage is characterized by a set of strategic imperatives and internal firm dynamics. The offshore insourcing is referenced in stage four. Carmel (2002) suggests that technology companies in stage four have different organizational structures and mechanisms. These firms have accumulated considerably more experience in offshore IT sourcing, but they usually preferred to own their IT units, sourcing from within their firms. The main characteristics of wholly-owned offshore units are:

- Sophisticated marketing efforts: since they are not part of the domestically located core IT function, they need an active and aggressive marketing strategy;
- Distinct organizational functions: some of these firms used their offshore IT units for software product R&D, internal information system work, and providing IT professional services to other firms. Some firms can support all types of software work, while others can support one or two;
- The value of vertical integration: these firms prefer the “build versus buy” argument, and they also prefer having vertical integration and an internal locus of control. Moreover, the ramp-up time is shorter because internal contracting is simpler, insourcing have advantages in the areas of security, confidentiality, and maintaining proprietary knowledge. Finally, internal IT professionals use standard software engineering tools, methodologies, and work process, which reduce the project management problem.

On the other hand, problems do exist and a comparison with other organizations in the same situation can improve the addressing of these problems. This is what we will do in our research.

RESEARCH METHOD

The first phase of this research is an exploratory study with the objective of increasing knowledge where typically practice has preceded theory. The main research method in this phase will be the case study (Yin, 2001). Figure 3 presents the main steps in the first phase.

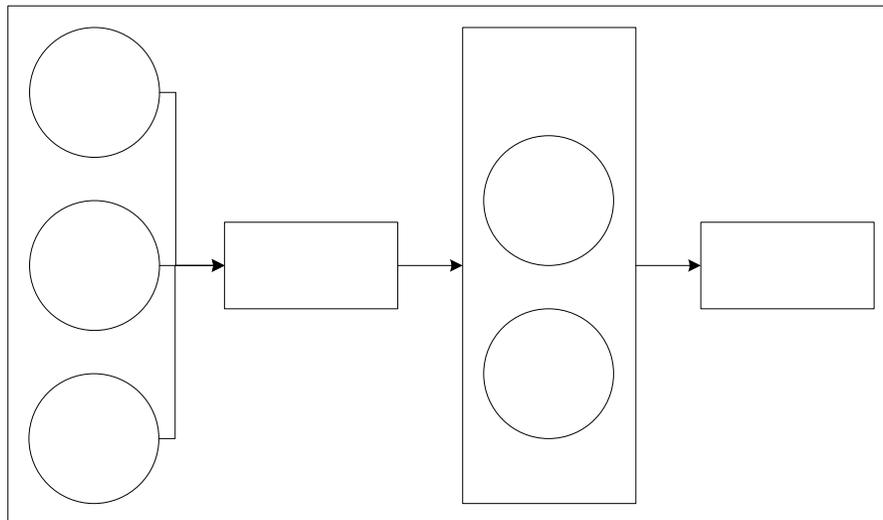


Figure 3. The research plan

The objective of the initial phase is to propose a preliminary maturity model of software development in offshore insourcing based not only on the extensive combined experience of the researchers in this area but also on a theoretical and empirical approach. Concurrently with a complete literature review, several case studies will be conducted using software projects as the unit of analysis. The next phase will include multiple case studies, starting the validation of the preliminary model as well as extending it.

These sets of case studies will be conducted using several data collection methods, allowing for triangulation of data and therefore increasing the results reliability, and quality. In particular, we will use interviews, semi-structured and cross-sectional surveys, and document analysis. This will be done in agreement with the approaches proposed by Yin (2001), and Krippendorff (1980). The last step is then to propose a Maturity Model of IT Distributed Projects in Offshore Insourcing Environments.

Future developments in this research line are planned to include quantitative confirmatory studies to fully validate the proposed model and map its characteristics in a global level. Benefits such as benchmarking possibilities can then be reached.

RESEARCH OBJECTIVES AND EXPECTED RESULTS

The objective of this research is to propose a maturity model for distributed IT projects for those organizations involved in offshore insourcing environments. To achieve this objective, we define the following specific goals:

- Learning from and improving the theory on offshore sourcing and DSD;
- To identify the main characteristics of the offshore insourcing environment;
- To develop a preliminary model that identifies the main DSD environments and their relationship with the offshore insourcing environment;
- To identify other aspects related to all stakeholders involved in the offshore insourcing.

Expected Results

At the end of the research, we expected the following results:

- The maturity model for distributed IT projects in offshore insourcing environments;
- A description of the different maturity phases;
- A “future view” of which issues to expect as one moves forward, and solutions that have worked for other organizations;
- The ability to benchmark the organizations against others;
- Development of a “best practices” body of knowledge;
- Development of action items that allow your organization to move faster to the more appropriate stage based on your needs and reality;
- Develop technical reports and papers to validate the model in the industry and academic communities.

CONCLUSION

This project is not only a landmark study in the area of offshore insourcing, something until recently not been researched, but also has strong implications to the more traditional offshore outsourcing. The key reason is that most of the work currently being done in offshore outsourcing is seen under the perspective of contracting; although obviously very relevant, eventually such studies will need to go further past that issue – which is exactly what we are proposing now.

REFERENCES

1. Carmel, E. (1999) *Global Software Teams: Collaborating Across Borders and Time Zones*. EUA: Prentice Hall.
2. Evaristo, J. R.; Scudder, R.; Desouza, K. and Sato, O. (2003) A Dimensional Analysis of Geographically Distributed Project Teams: A Case Study, *Journal of Engineering Technology and Management*, 2003.
3. Herbsleb, J.; Moitra, D. (2001) *Global Software Development*. IEEE Software, California, v.16, n.2, p. 16-20, Mar /Apr.
4. Karolak, D. W. Global Software Development (1998) *Managing Virtual Teams and Environments*. Los Alamitos, EUA: IEEE Computer Society, 1998.
5. Krippendorff, K. (1980) *Content analysis: an introduction to its methodology*. Sage.
6. Prikladnicki, R.; Audy, J. L. N.; Evaristo, R. (2003) Distributed Software Development: Toward an understanding of the relationship between project team, users and customers. *Proceedings of ICEIS*, Angers, France.
7. Yin, R. (2001) *Case Study: Planning and Methods* (in Portuguese), São Paulo, Bookman.
8. Carmel, E.; Agarwal, R. (2002) The Maturation of Offshore Sourcing of Information Technology Work, *MIS Quarterly Executive*, Vol. 1, No. 2, June 2002, 65-77.
9. Damian, D. (2002) The study of requirements engineering in global software development: as challenging as important, *Proceedings of International Workshop on Global Software Development at ICSE*, Florida, USA.

ⁱ Research developed at CDPe research center at PUCRS University, and funded in terms of the Brazilian Federal Law for IT.