Managing outsourced software development: Does organisational distance demand different project management?

Tom McBride, Brian Henderson-Sellers, Didar Zowghi
Department of Software Engineering
Faculty of Information Technology
University of Technology, Sydney
{mcbride, brian, didar}@it.uts.edu.au <mailto:mcbride, brian, didar@it.uts.edu.au}

Abstract

Software development is increasingly spread around the world through partnering agreements, distributed development centres within multi national organisations and outsourced development. Increased organisational distance, intuitively, would affect the way in which distant project tasks were monitored and managed. To investigate whether and in what way project management is affected by organisational distance, this paper first examines existing theories to establish expected variations. Hypothesised differences are examined through a research project that investigated, among other things, task control and organisational distance. The research found that, contrary to expectations, organisations do not alter their project management processes as organisational distance increases.

Introduction

Distributed and outsourced software development projects are becoming more common and while there have been some experience reports that suggest such projects must be managed differently, there is little specific advice on just what those differences might be. Much of the available guidance to project managers, notably the PMBOK as well as many common texts (Burke, 2003; Cleland & Ireland, 2002; Hughes & Cotterell, 1999; McConnell, 1998), presents project management as if the project was entirely co-located and where direct and informal monitoring is possible. However, when a development team is separated from the project manager, either geographically or organisationally, such direct and informal monitoring is less possible. Carmel (1999) identifies five "centrifugal forces" of global software teams, any one of which has the potential to seriously jeopardise a project's success.

This paper first examines the general problem of managing at a distance to establish whether and how organisational distance affects how tasks are monitored and managed. This is then related to software development tasks and a hypothesis is proposed that software development tasks are affected by organisational distance. Empirical evidence is examined to support or refute the hypothesis and conclusions drawn. Finally, the significance of the findings is summarised.

Related work

The problem of managing at a distance has been investigated from a number of disciplinary viewpoints and this paper will examine three such approaches: *agency theory*, *cultural distance* and *organisation theory*. From these different approaches a hypothesis will be drawn regarding a relationship between organizational distance and choice of method to control software development tasks.

Agency Theory.

The original ideas of agency theory were proposed as a broadening of the risk sharing literature to include the so-called agency problem when cooperating parties have different goals and different division of labour (Eisenhardt, 1985, 1989; Hamilton & Kashlak, 1999). Agency theory is directed at the relationship where one party (the principal) delegates work to the other (the agent) who performs that work (Eisenhardt, 1989). Control Theory (Henderson & Lee, 1992; Kirsch, 1996) examines the same problem using slightly different terminology. While both Agency Theory and Control Theory originated from applying the principles of control systems to organisations, Agency Theory is directed at the specific problem of principal-agent relationships whereas Control Theory is concerned with the general problem of control within organisations.

Both Agency Theory and Control Theory predict that the type of system appropriate for controlling a particular task depends on task programmability and output measurability. Briefly, task programmability is "the degree to which appropriate behaviour of the agent can be specified in advance" (Eisenhardt, 1989). The same attribute is approached from a different perspective by Control Theorists who refer to "Knowledge of the transformation process" being "the organisation's understanding of how inputs are converted into outputs" (Hamilton & Kashlak, 1999).

Agency Theory and Control Theory identify several types of control that can be used in differing circumstances. The control methods are classified according to which part of the task, its inputs, behaviours or outputs, are controlled.

Behaviour control

Behaviour control is exercised through procedures. Rewards are given for following the procedure regardless of the outcome. This form of control is appropriate when the task can be fully specified. The well known example of fast food outlets tend to follow this control method. It doesn't make much sense to invest heavily in training the staff when job rotation and staff turnover means they are unlikely to stay at the one job for several years. But it is comparatively easy to specify exactly how the food is to be prepared and cooked, and train the staff to follow the procedure.

Outcome control

A task where the principal has very little knowledge of the transformation process, or the transformation is difficult to specify, tends to rely more on outcome control. That is, specifying the results of the transformation process rather than the process itself. The information required by outcome control is considerably less and generally easier to measure and collect than the information required by behaviour control.

Input control

Sometimes it is difficult to control a task through measuring the outcomes and difficult to prescribe the task behaviours. For example, though we may recognise a 'good system design' it is not easy to prescribe a series of actions that would result in a 'good system design'. Rewarding someone for developing a 'good system design' won't necessarily produce a one. In such circumstances, task control is exercised by recruitment, training and selection of those who will carry out the task. Control is exercised through controlling the inputs to the task, rather than the behaviours of the task or the task outputs.

Relationships between control methods

There is a relationship between the control methods outlined. The relationship is stated by Hamilton and Kashlak (1999) as follows;
- If behaviour can be specified and monitored, behaviour control is likely to be favoured.
- If behaviours are difficult to specify or monitor but outcomes can be specified and measured, outcome control is likely to be favoured.
- If behaviours are difficult to specify or monitor and outcomes are difficult to specify or measure, input control is likely to be favoured.

Input control is the preferred control method only when cultural distance was considered. Then if task programmability and output measurability were both low, there is a higher probability that input control will be favoured.

The issues of task measurability, knowledge of the transformation process (behaviour) and the cost of verifying behaviour compliance will be examined in the following sections.

Transformation processes

The transformation processes of software development are those collective processes that convert the requirements into the deliverable software. Such

processes have been developed, published and debated since software's earliest days to the point where there is sufficient agreement on the broad thrust of the processes to have ISO standards (ISO 12207, 2002; ISO 15288, 2003), commercially available methodologies and healthy debate over which of several software development methods suit different applications. It is reasonable to conclude that software development processes (transformation) are reasonably well understood but not necessarily followed.

Transformation processes, embodied in software development process descriptions, describe required behaviours and are commonly used to enforce behaviour control.

Output measurability.

The output of a software development task is usually a deliverable software component or system that can, in theory, be measured against its requirements by testing. Some years ago it was considered that a software product could only be tested, or evaluated, for functional requirements. But considerable progress has been made in establishing frameworks for evaluating non-functional requirements, reliability and dependability for example, and the methods by which they might be evaluated (M.R. Lyu, 1994; Michael R. Lyu, 1996; Musa, 1997; Wattanapongsakorn & Levitan, 2000). In the context of agency theory, the measurability of the output of a software development task is directly related to how well it can be evaluated against its requirements. For a small component with few behaviours and tightly specified interfaces, evaluation is likely to be relatively straightforward; a large, complex system is much more difficult to evaluate.

Verification cost.

Part of the task of software development project management is gathering the information about the progress and state of a project. That is, are people performing those tasks assigned to them as scheduled and are the expected outputs being produced? This information gathering will obviously be affected in some way by the organisational distance between the source and consumer of the information. If they are co-located then the difficulty, and costs, will be minimal. But if the two belong to separate organisations and are separated by time zone, language and culture then not only will communication logistics be much more difficult, but seeking specific information and understanding the presented information will also be more difficult (Carmel, 1999).

Cultural distance should increase the cost of gathering information and would, according to agency theory, tend to favour outcome control over behaviour control. This should be evident for all organisation processes including those relevant to project managing software development.

Organizational distance

Separation between the project manager and elements of the project team could be measured in a number of ways. For this research, a measure of separation was needed that incorporated the different factors that contribute to increasing difficulty that the project manager may experience when trying to manage remotely. Such a measure was proposed by Napier and Ferris (Napier & Ferris, 1993) to reason about separation between supervisor and subordinate. However, this research could not assume that the same supervisor - subordinate relationship operated. Thus, a new measure of organizational distance was developed (McBride, 2005), based on that of Napier and Ferris which used the three dimensions of psychological distance, structural distance and functional distance. The new measure retained the thrust of Napier and Ferris's measure but used factors that could apply more easily to globally distributed projects. Psychological distance became cultural distance. Structural distance was retained. Functional distance became administrative distance, a measure of the administrative separation between the project manager and the team member. The advantage of the new measure was that, for the purposes of this research, it was not difficult to develop ordinal measures for each dimension, and not difficult to combine the dimensions to derive an overall ordinal measure of organizational distance.

There have been some experience reports (Borchers, 2003; Nicholson & Sahay, 2001) and empirical studies (Howell, Romero, Dorfman, Paul, & Bautista, 2003) concerning the effect of cultural distance on project management with mixed results. The experience reports tend to conclude that cultural distance makes management more difficult whereas the empirical report concluded that the nationality mattered less than the personal characteristics of the manager.

Studies concerning structural, largely geographical, separation between parts of a project team have concluded that separation causes delays (Herbsleb, Mockus, Finholt, & Grinter, 2000, 2001). Other studies (Cramton, 1997; Jarvenpaa & Leidner, 1998) involving both cultural separation and structural separation indicate that there are many obstacles to communication, with separation being only one of them.

There have been few recent studies concerning administrative distance. Nevertheless, studies concerning organization structure changed to reduce distance between decision maker and those carrying out the work (Lawrence & Lorsch, 1967; Perrow, 1986; Woodward, 1958). A more recent study (Berggren, Soderlund, & Anderson, 2001) confirmed the tendency for increased administrative distance to be accompanied by increased management difficulties.

Different circumstances may favour a different combination of factors to indicate organizational distance between people in different parts of an organization, but for this research the factors presented served the purpose well.

Theoretical conclusions

Of these three approaches, agency theory and cultural distance predict that an organisation's project monitoring and management processes will be affected by organisational distance, while organisation theory, specifically Mintzberg's model of organisations, predicts that there will be a change only under specific circumstances.

To investigate the question of whether or not project monitoring and management processes differed as organisational distance increased the following hypothesis, with associated null hypothesis, was proposed.

H1: The use of outcome control is more likely as organisational distance increases.

H0: The use of outcome control is unrelated to organizational distance.

For software development, the visible signs of outcome control would be an emphasis on evaluating the completed product - testing. Behaviour control would be evident from requiring the outsourced organisation to adopt standard software development and project management processes, together with some regular project monitoring or reporting.

Research method.

Structured interviews were conducted with project managers from software development organisations in Sydney, Australia between February and September 2003. Organisations were approached initially by phone and asked if there was a project manager involved in software development and willing to be interviewed. Structured interview allowed questions and responses to be clarified or amplified during the interview and also allowed for unexpected information and findings to emerge rather than directing responses to preconceived models.

There were 49 questions. Of these
- 4 questions categorised the organisation and its software development processes,
- 7 questions established how the project manager monitored the project,
- 3 questions established how the organisation adjusted the project (scope, schedule, quality requirements, performance requirements) as a consequence of monitoring the project,
- 8 questions established an approximate measure of organisational distance,
- 5 questions established project monitoring processes for outsourced tasks and
- 2 questions established how the outsourced project tasks were managed in response to information revealed by project monitoring.

An expected range of responses was developed for each of the 49 questions, both to guide the questioning and responses, and to help guide later analysis. These range of responses were not shown to the interviewee and were used to indicate the scope of the information sought. For example, one of the questions and the expected responses was

> *Is there a standard method or process for monitoring project tasks?*
> - *No - each PM does their own thing.*
> - *Yes, but informal and flexible.*
> - *Yes, defined but not very extensive.*
> - *Yes, defined and extensive.*

This reduced the tendency to, for example, try to describe the project monitoring method when all that was sought was whether or not there was a standard method and to get some indication of its rigour.

Questions were generally of two types. The first was intended to establish the organisation's position on some scale. For example, a question about the size of the organisation was intended to establish if they were small, medium, large or multinational. The second type of question was more open and designed to elicit information on, for example, the range of subjects discussed at an internal project meeting or the types of adjustments made to a project schedule in response to events. This type of question in surveys would usually have place to respond with "Other" and space for the respondee to expand on "other".

Each interview took between 30 minutes to just over an hour, depending on the loquaciousness of the interviewee. Most lasted about 45 minutes. The interviews were conducted at the interviewee's worksite. They were audio taped and later transcribed. The transcription was sent back to the interviewee to check and correct. Interviewees were told that the objective reviewing the transcript was to check that they were happy with the responses and to change them if they were not, rather than to check that it was an accurate transcript of the interview. The interview responses were then encoded and analysed using SPSS 11.0. Since the encoded variables were nominal or ordinal, the only statistical analysis was Chi-square if both variables were nominal type, and Kendall's tau-b if one of the variables was ordinal type.

Sample characteristics.

Organisational size

Organizational size was judged largely on the number of personnel. This estimate included the whole organization, not just the software development part of it because past experience indicates that a small division within a large organization more closely resembles the large organization than a small, independent company of similar size to the division. Table 1 gives the distribution of organization size. The size divisions were chosen because they reflect approximately where organizations tend to change structure, from direct supervision through simple, single layer management through to multi layer management.

Table 1: Organisational size

Small (< 30 staff)	12
Medium (31 - 120)	4
Large (>120 - 1000 single organization)	3
Multinational (> 1000 or Multinational)	12

Process maturity

The process maturity is a very approximate guide based on the ISO 15504 (SPICE) or CMMI scale of process maturity. The first author is familiar with, and practised at, such process assessments and claims these ratings would be the equivalent of a very low rigour SPICE assessment. The single instance of a maturity level of 5 (Table 2) came from an organisation that had recently undergone a CMMI assessment and was accredited with that level.

Table 2: Process maturity

Informal - Level 1	6
Managed - Level 2	8
Defined - Level 3	16
Measured - Level 4	0
Optimizing - Level 5	1

Organisations were adjudged at level 3 if they were ISO 9001 accredited or had undergone a SPICE or CMMI assessment and had achieved that rating. Level 2 was assigned if the organisation had documented software development processes, particularly those dealing with project management and document control.

Organisational distance

Organisational distance was assessed from a combination of geographical distance, organisational separation and cultural distance. Organisational separation was close if the two belonged to the same organisation, medium if they belonged to separate divisions of the same organisation such as different divisions of a multinational and distant if the two organisations were completely separate. Cultural distance was taken from the scale of cultural distance developed by Hofstede (Hofstede, 1983; 1991). Development being performed in a country such as USA, Canada or UK by a different organisation was judged to be a medium organisational distance. Development performed by a division of the same organisation in India or Japan was considered medium-distant. There was no more distant development than that in this sample. Eight organisations developed entirely locally and did not outsource any development.

Table 3: Organisational distance

Close (co-located)	5
Close - Medium (same organization, same city)	6
Medium (Different organization, same country or same organization, different country)	9
Medium - Distant (Different country, different organization, both software developers)	3
Distant (Different country, different organization, different field)	0

Outsourcing

Of the 31 organizations interviewed, 23 undertook some form of outsourcing (Table 4). This ranged from simply hiring contractors or contracted companies who worked as if they were employees, through to the more expected form of outsourcing by which means another company is contracted to develop software, sometimes in another country. Sometimes the development teams were made up of the organisation's employees, some employees from the customer's organisation and some independent contractors. Such "blended teams" were managed for the duration of the project and in matters related to the project as if they were all employees of the organisation. Sometimes such teams were co-located and sometimes not.

Table 4: Degree of outsourcing

No outsourcing	8
Contractors	9
Full outsource	14

Findings

There was some anecdotal evidence that the costs, both financial and logistical, of seeking the same information for a development being carried out at a distance were higher and accepted as inevitable, as project managers held conference calls during inconvenient times or travelled to the other country to seek information in person.

There was no correlation between organisational distance and whether or not the outsourcee was trained in the organisation's development methods, nor between organisational distance and the amount of responsibility given to the outsourcee

for software component design. There was also no correlation between organisational distance and the development stage at which work was checked against its requirements. By implication since the development stage determines the method used to check the requirements, there was no correlation between organisational distance and the method used to check a developed product against its requirements.

Similarly, there was no correlation between the organisation's process maturity and the same factors of training and output checking against requirements. There was a correlation between process maturity and design responsibility (Pearson's Chi-square Asymptotic Significance of 0.004).

The same three factors (training, design responsibility and output checking stage) showed no correlation with the degree of outsourcing. This could due to any of several reasons. It could be an artefact of the research sample being too small to show statistical differences. It is more likely that organisations have not yet encountered significant problems that are obviously attributable to the degree of outsourcing and don't have any reason to change.

Similar analyses of organisational size against training, design responsibility and the output checking stage did not establish any correlation. While it is possible that training, design responsibility and output checking are correlated with organisational distance but dependent on organisational size or process maturity, the research sample was too small to detect any statistical effect.

Specific questions were asked about communication difficulty due to language or culture. Some respondents reported degrees of communication difficulty but statistical evidence of a correlation between communication difficulty and organisational distance was weak. Pearson's Chi-square test significance was 0.342 while Kendall's tau-b significance was reported at 0.033. If further research on this specific issue was conducted and affirmed a correlation stronger than this, then it would support a hypothesis that organisational distance increased project monitoring costs. This could lead to a model of project management that predicted which type of project monitoring, or control method, would be most economic and effective for differing organisational distance.

There was also some anecdotal evidence that the type of task affected the control method. When the task was installation related rather than strictly software development, some organisations placed a lot of emphasis on training. This would indicate that input control was more likely to be associated with tasks like system installation or implementation where the outcomes are more varied and possibly harder to measure.

Threats to validity

Small sample size. The sample was relatively small at 29 and many statistical tests suffered from having insufficient cell counts, usually less than 10.

Non random sample. The participating organisations were those listed in the Sydney, Australia, Yellow Pages who agreed to be interviewed when approached by telephone. Soliciting started at the beginning of the list of those organisations

listed under "Computer Software and Packages" and proceeded until sufficient data had been gathered to provide a useful, if limited, source. Such *accidental sampling* is considered to have very weak external validity and likely to be biased (Trochim, 2001).

Weak external validity. Organisations with low maturity, chaotic project management processes are less likely to be willing to reveal to a researcher just how they manage projects, or don't manage them. Consequently the findings of this research are likely to be biased toward the more mature organisations. However, given the conclusions, the weak external validity is of less importance.

Localised sample. The research sample was taken from organisations in Sydney, Australia. While there were a significant number of multinational organisations in the sample, it is possible that the same research findings are similarly localised. The study would need to be replicated in another country to test this.

Conclusions

The null hypothesis, that outcome control is unrelated to organisational distance, is accepted. Processes used by an organisation to monitor and manage software development tasks were not found to differ in practice according to their distance from the organisation. Organisations, at least those that participated in this research, seem to adopt a particular process then apply it to all situations. If project managers were more able to articulate what signs of progress and project health they sought through their monitoring they may be able to devise and use a project monitoring process that relied less on their active involvement and more on objective project measures.

It is possible that project managers use the same processes regardless of organisational distance because they are not aware of any alternative methods. In which case, as awareness grows of alternatives, project managers may well adopt differing processes according to the circumstances.

Further research is needed to investigate the weak correlation between organisational distance and costs of project monitoring and management. If such a correlation exists, then there is likely to be a point at which the incumbent process becomes uneconomical compared to alternative control methods.

Similarly, the anecdotal connection between task type and control type needs further investigation. As the software development processes become more fragmented and organisations specialise in, for example, software development and outsource testing or installation, there is a danger that they may employ an inappropriate control method.

Although national and organisational cultures can't be ignored, their effect on software development project management methods currently appears to be minimal. The same methods appear to be effective across wide cultural distances. This implies that an organisation contemplating outsourced software development need not change its project monitoring and management process for reasons of organisational distance. There may be other reasons to change such as cost or the nature of the outsourced task, but not distance.

Acknowledgments

References

Berggren, C., Soderlund, J., & Anderson, C. (2001). Clients, contractors, and consultants: The consequences of organizational fragmentation in contemporary project environments. *Project Management Journal, 32*(3), 39-48.

Borchers, G. (2003). *The software engineering impacts of cultural factors on multi-cultural software development teams.* Paper presented at the Proceedings of the 25th international conference on Software engineering, Portland, Oregon.

Burke, R. (2003). *Project Management: Planning and Control Techniques.* (4th ed.). Tokai, South Africa: Burke Publishing.

Carmel, E. (1999). *Global Software Teams: Collaborating Across Borders and Time Zones.* Upper Saddle River: Prentice Hall PTR.

Cleland, D. I., & Ireland, L. R. (2002). *Project management: Strategic Design and Implementation* (4th ed.): McGraw-Hill.

Cramton, C. D. (1997). Information Problems in Dispersed Teams. *Academy of Management Proceedings*, 298-302.

Eisenhardt, K. M. (1985). Control: Organizational and Economic Approaches. *Management Science, 31*(2), 134-148.

Eisenhardt, K. M. (1989). Agency Theory: An Assessment and Review. *Academy of Management Review, 14*(1), 57-74.

Hamilton, R. D., III, & Kashlak, R. J. (1999). National Influences on Multinational Corporation Control System Selection. *Management International Review, 39*(2), 167-189.

Henderson, J. C., & Lee, S. (1992). Managing I/S Design Teams: A control Theories Perspective. *Management Science, 38*(6), 757-777.

Herbsleb, J. D., Mockus, A., Finholt, T. A., & Grinter, R. E. (2000). *Distance, dependencies, and delay in a global collaboration.* Paper presented at the Computer Supported Cooperative Work, Philadelphia.

Herbsleb, J. D., Mockus, A., Finholt, T. A., & Grinter, R. E. (2001). *An empirical study of global software development: distance and speed.* Paper presented at the International Conference on Software Engineering, Toronto.

Hofstede, G. (1983). National Cultures in Four Dimensions: A research based theory of cultural differences among nations. *International Studies of Management and Organization, 13*(1-2), 46-74.

Hofstede, G. (1991). *Cultures and Organizations: Software of the Mind.* New York: McGraw-Hill.

Howell, J. P., Romero, E. J., Dorfman, P. W., Paul, J., & Bautista, J. A. (2003). Effective leadership in the Mexican maquiladora: challenging common expectations. *Journal of International Management, 9*(1), 51-73.

Hughes, B., & Cotterell, M. (1999). *Software Project Management*. London: McGraw-Hill.

ISO 12207. (2002). Information technology -- Software life cycle processes - Amendment 1, *12207*.

ISO 15288. (2003). Systems engineering - System life cycle processes, *15288*.

Jarvenpaa, S. L., & Leidner, D. E. (1998). Communication and Trust in Global Virtual Teams. *Journal of Computer-Mediated Communications, 3*(4).

Kirsch, L. J. (1996). The Management of Complex Tasks in Organizations: Controlling the Systems Development Process. *Organization Science, 7*(1), 1-22.

Lawrence, P. R., & Lorsch, J. W. (1967). High-performing Organizations in Three Environments. In D. S. Pugh (Ed.), *Organization Theory: Selected Readings* (2nd ed.): Penguin Books.

Lyu, M. R. (1994). *Software reliability: to use or not to use?* Paper presented at the 5th International Symposium on Software Reliability Engineering, Monterey, CA.

Lyu, M. R. (Ed.). (1996). *Handbook of Software Reliability Engineering*. New York: McGraw-Hill.

McBride, T. (2005). *The use of project management mechanisms in software development and their relationship to organizational distance: An empirical investigation*. PhD Thesis, University of Technology, Sydney, Australia.

McConnell, S. (1998). *Software Project Survival Guide*: Microsoft Press.

Musa, J. D. (1997). *Introduction to software reliability engineering and testing*. Paper presented at the The Eighth International Symposium on Software Reliability Engineering, Albuquerque, NM.

Napier, B. J., & Ferris, G. R. (1993). Distance in Organizations. *Human Resource Management Review;, 3*(4), 321-357.

Nicholson, B., & Sahay, S. (2001). Some political and cultural issues in the globalisation of software development: case experience from Britain and India. *Information and Organization, 11*(1), 25-43.

Perrow, C. (1986). *Complex Organizations: A Critical Essay* (Third ed.). New York: Random House.

Trochim, W. M. K. (2001). *The Research Methods Knowledge Base*. Cincinnati: Atomic Dog Publishing.

Wattanapongsakorn, N., & Levitan, S. (2000). *Integrating dependability analysis into the real-time system design process*. Paper presented at the Reliability and Maintainability Symposium, 2000., Los Angeles, CA.

Woodward, J. (1958). Management and Technology. In D. S. Pugh (Ed.), *Organization Theory*. London: Penguin Books.