

Distributed Inter-Organisational Outsourcing: A Case Study

Seán Linnane, Ita Richardson

Department of CSIS and Lero – The Irish Software Engineering Research Centre
University of Limerick, Ireland
sean.linnane@ul.ie, ita.richardson@ul.ie

Abstract

As more and more software organisations continue their integration into a burgeoning global business community, the effective management of relatively new types of working arrangements, such as those enabled by virtual teams and distributed software development (DSD), assumes heightened importance. Membership of this community promises increased opportunities along with well-publicised risks. The ability to harness the capabilities of people, process and technology in a distributed environment is, however, considered a greater challenge than in its co-located equivalent. Although pertinent to software vendors of all sizes, the literature to date has primarily concentrated on DSD from the perspective of larger organisations. Paradoxically, however, the majority of organisations developing software globally are small in size.

This study seeks to restore the equilibrium somewhat by using a case study approach to investigate DSD in the context of a small software provider based in Ireland. The findings emphasise particular difficulties in relation to communication, people management, trust, skills and training, and the lack of standard or consistent process, and describes how the company is dealing with them. Appreciating and understanding these difficulties can help small companies and clients alike to better prepare for the challenges of DSD and outsourcing.

1.0 Introduction

Distributed Software Development (DSD) is a phenomenon of escalating significance for organisations competing in today's highly globalised business landscape [1, 2, 3]. While many organisations, particularly larger ones, are actively and intentionally securing membership of this global community, many more are assuming and inheriting citizenship on an involuntary basis. In this regard, the software industry offers no exceptions with many organisations benefiting from geographic presence in far-flung regions across the world while others just compete within local, regional and national ecosystems.

Much of the research and evidence reported in the literature with regard to DSD is laden with reports and experiences of those working within the confines of large organisations [4, 5, 6]. This is understandable, given that large companies by their nature are generally comprised of a much larger workforce and also tend to have significantly more financial reserves at their disposal than their smaller counterparts. Furthermore, it is not uncommon for large companies to have offices and centres dotted in multiple geographic regions, both at a national and international level. On the other hand, however, this disproportion in terms of reporting represents a dichotomy in the sense that the majority of companies developing software around the world are considered small in size [7]. The corpus of literature is far from clear as to what exactly constitutes a “small” company. While geographic location and other criteria such as sales revenue, total assets, and the value of owners’ equity are often used to distinguish small companies from their larger counterparts, the number of employees is generally the most popular measurement used [8]. For the purposes of this paper, small companies are defined as those comprising a workforce of up to fifty employees [9].

This paper presents data elicited from a case study within a small indigenous software provider in Ireland. We studied four inter-organisational software development projects within this company. The company, ABC (a fictitious name), employ a total of forty-nine employees. Of these, thirty-four are involved in software development. While the company are currently involved in inter-organisational DSD within Ireland, future plans include expansion into the U.K. market, where they will continue to distribute their development. This paper highlights a number of challenges experienced by ABC within a distributed and outsourcing context. From an academic perspective, this paper contributes to the lack of literature focusing on smaller companies operating within such a context. While fully acknowledging the danger in premature generalisation of case study findings, nonetheless it is anticipated that the findings outlined here will be of interest to other small software providers operating within a nationally distributed and inter-organisational software development environment.

The remainder of this paper is organised as follows: The next section synopsis related work in the field of DSD. Section 3 sketches the context by profiling the company at the centre of this study. Section 4 articulates the chosen research methodology. The penultimate section describes the challenges experienced by the investigated company and the measures used to deal with those challenges. Finally, the paper summarises those challenges and proposes future work.

2.0 Distributed Software Development (DSD)

In the literature, the term “Distributed Software Development” is used interchangeably with the term “Global Software Development” (GSD) and “multi-site development”[2]. However, “GSD” tends to imply the focus is exclusively or predominantly on software developed on an international scale, that is, between and across companies in different countries and continents. As a result, this paper

prefers to use the acronym DSD as the company at the centre of this case study was, at the time of data collection, developing software on a national scale only.

Proponents point to many compelling incentives for tending towards a DSD mode of software development. These range from labour costs and skills arbitrages at one end of the spectrum to greater market presence, growth and diversification at the other end. The potential to achieve decreased time-to-market, enabled by 'round-the-clock' development, is also a much-heralded advantage touted by advocates of DSD[1].

To fully avail of these benefits, however, those interested in DSD must take cognisance of a vast array of well-documented challenges, including those of a technical, social and process nature. Despite best efforts, history has demonstrated that the effective management of co-located software development work has and continues to prove difficult to master. However, today's contemporary business landscape puts increasing demands on the practice of developing software as more and more work begins to traverse organisational and geographic boundaries.

Both practice and theory point to communication, collaboration, coordination and control as fundamental processes underpinning all DSD engagements. Even though issues pertaining to these processes equally apply to their co-located equivalents, there is universal consensus that geographic distance exacerbates the difficulties associated with their control and management [2, 10]. Other challenges reported include the reliance on asynchronous modes of communication [11], a lack of trust and openness, inappropriate processes and procedures [10], and unclear roles and responsibilities. Furthermore, there is already considerable evidence [12, 13] to suggest that culture is an integral piece in the DSD jigsaw as participants regularly come from very disparate backgrounds with very opposing values and beliefs.

It is clear, therefore, that despite the many potential benefits accruing to it, DSD is nonetheless an extremely complex and multi-faceted undertaking. Being aware of and understanding the many nuances is a necessary first step in implementing a successful DSD strategy.

2.1 Software Outsourcing

Software outsourcing, one particular variation of DSD, is defined as the transfer of responsibility for IT/IS assets, resources and/or activities (e.g., planning, management and operations) to an external service provider [14, 15]. This definition suggests the existence of some form of inter-organisational business relationship between two distinct organisations (a client organisation and a software provider/vendor) to address some specific IT/IS requirements for the client organisation.

While outsourcing demonstrates many of the characteristics discussed previously, there exists a further difficulty in managing outsourced projects due to the differences in organisational goals and structures between the provider and the client, distance between stakeholders, as well as the inability to use hierarchies

which help in internal IS development [16]. Therefore, companies who embark on outsourcing are faced with additional complications to the challenges of DSD.

3.0 Research Project

The purpose of this research was to examine the challenges facing small software companies involved in DSD and outsourcing relationships. To accomplish this, a case study was conducted with one such company, ABC (a pseudonym). This section describes the context in which the research was undertaken.

3.1 Context – Profile of ABC

Company ABC is a small software company with its headquarters in Dublin, Ireland and its software development centre situated 200 kilometres away in the Mid-West region of the country. Primarily specialising in the provision of outsourced application development and support services across a wide spectrum of business sectors and industries, ABC's client base mainly comprises companies within the financial services, telecommunications and government sectors. In addition to technical expertise, the company also offers a business consultancy service to clients. Just over a third of its overall workforce of forty-nine is based in its software development centre, with eleven practitioners employed as software contractors.

3.2 Why Multi-Site?

To achieve their business goals, management at ABC took the strategic decision to partition its two core services, consultancy and outsourcing, across two separate sites. As virtually all of the companies targeted by ABC were based in Dublin, it made sense to establish its main hub nearby. From here, ABC offer management and consultancy services within convenient proximity of their client base.

After assessing their options, management decided to locate its software development centre in the more cost-effective Mid-West region of Ireland. The rationale for establishing its development centre there was based on more than economic reasons alone, and included:

- Eligibility for grant-aid and funding
- Sound transport and communications infrastructure
- Access to third-level graduates and research
- Availability of personnel with product expertise

However, as discussed later, the physical separation between the development centre, ABC's headquarters and their clients introduced unanticipated complexities.

3.3 Project Stages by Location

In general, ABC conducts the initial stages of software projects at the client's site. Typically, this encompasses analysis and preliminary design work. One reason for this approach is that the co-location of project resources provides a vehicle for forging strong work relationships and helps to promote trust-building, a key

enabler of harmonious and successful work relationships [2, 3]. As well as allowing more convenient access to subject matter experts, business users and other project stakeholders, such an arrangement facilitates more informal communications [11]. In contrast, the detailed design, development, and preliminary testing are usually conducted at ABC's development centre.

3.4 The Use of Contractors

Due to limitations in terms of size and expertise, ABC relies on the help of external software contractors to provide services to its clients. At the time of this study, eleven practitioners were employed in a contractor capacity, representing almost one third of ABC's total software development workforce. Although this enabled ABC to overcome the technical nuances of the projects, it also introduced yet another layer of complexity in the overall management of the projects, as described in Section 5.

4.0 Research Methodology

The purpose of this study was to explore and describe the difficulties facing small vendors operating within a distributed software development environment, and particularly in the context of inter-organisational software outsourcing. For this reason, a case study approach was deemed appropriate as the primary research methodology for this project. A case study in terms of the research process, is defined as "*...an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident*" [17]. This definition suggests the case study method is particularly adept at revealing and illuminating the relationship between the phenomenon under investigation and the context in which it is occurring [18]. Such an approach is considered appropriate if the focus of the study is to investigate the 'how' and 'why' of a phenomena within its organisational context [19]. Furthermore, a case study approach incorporating in-depth interviewing is deemed appropriate for research which has a descriptive and exploratory focus [20].

Rather than guide the research through a set of preconceived and stringently specified research questions formulated exclusively from the viewpoint of the authors, a primary aim of this paper was to garner detailed and in-depth data from practitioners' perspectives. As a research method, interviews provide an appropriate vehicle for satisfying these requirements. However, as the most pertinent issues and concerns facing practitioners at ABC could not realistically be enumerated fully in advance, it was imperative that some latitude existed in terms of the formulation and sequencing of questions to be asked. The flexibility offered by a semi-structured interviewing approach complies with this goal by being more amenable to the emergence of alternative avenues of enquiry [18]. Such an approach makes it possible to frame successive questions according to preceding responses, to receive instantaneous clarification of any ambiguities, to probe unanticipated responses, and to be receptive to non-verbal cues. This contrasts acutely with more rigid and less interactive approaches to data collection which

often lack the texture and richness required for a more insightful appreciation of the investigated situation [21].

To facilitate interviews, an interview guide was devised outlining a preliminary list of areas to be addressed [18]. As a starting point, interviewees were asked to describe their experiences with the projects they were involved in. They were encouraged to elaborate on the difficulties they encountered, their understanding of why these problems were arising, and how, in their opinions, these difficulties were or could be alleviated. To augment this evidence, data collection also incorporated document examination in tandem with non-participant observation. Company-, project-, and process-related documents were identified and reviewed to attain a greater comprehension of projects investigated as part of this research. However, observation of participants in their natural habitat did not contribute greatly to data collected for a number of reasons. Very limited access was given to participants as they were very busy on a number of projects. Also, many participants seemed rather uneasy with the unintentional encroachment into their personal work space, thus compromising data quality. To clarify and refine issues emerging from the data, informal interviews and discussions were used.

In total, four projects were studied, and personnel at different levels (from software engineers to high-level management) were interviewed. Details are presented in Table 1. Further triangulation of data was achieved by interviewing ABC's Service Delivery Manager (SDM), who held overall responsibility for the development centre. The data generated from this research was analysed using content analysis [21]. The next section discusses the most salient points emanating from the analysis of the data.

Client	Industry Sector	Project	Delivery Model	Technology	ABC Project Manger	ABC Software Engineers
C1	Mobile Telecoms	P1	Distributed Team	Microsoft .NET	PM1	P1-SE1 P1-SE2
C2	Telecoms	P2	Distributed Team	Web Services, Mainframe	PM2	P2-SE1 P2-SE2
C3	Public Sector	P3	Distributed Team	Oracle Forms	PM3	P3-SE1 P3-SE2
C4	Financial Services	P4	Distributed Team	Oracle, Mainframe	PM4	P4-SE1

Table 1: Summary of projects investigated at ABC

5.0 Distributed Development at ABC

Upon analysing the output from the case study, a number of recurring issues were revealed. In particular, the findings emphasise difficulties with communication, people management, trust, skills and training, and the lack of standard or consistent process. The following subsections describe difficulties experienced by staff at ABC with respect to each of these five factors, while also outlining the ways in which they are being addressed.

5.1 Communication

The management of project communication is concerned with the timely and appropriate generation, collection, distribution, storage, retrieval, and ultimate disposition of project information [22]. A common pattern adopted at the outset of each project undertaken by ABC is to bring together stakeholders from the client company and team members from ABC assigned to the project. Aside from discussing the background to the project and its scope, this get-together also provides an opportunity to articulate the expectation of the projects and briefly discuss its technical aspects. These meetings often represent the first opportunity for all stakeholders to meet. Detailed requirements analysis and technical workshops frequently provide the next opportunities for many of these stakeholders to meet. Further face-to-face meetings often coincide with major project milestones and provide a convenient forum for identifying and discussing future business needs.

Projects falling under the remit of this study relied heavily on telephone and email usage as the primary modes of communication. In each of the four investigated projects, remote delivery was made possible using virtual private networks (VPNs). Apart from controlling access to the clients' technical environments, VPNs also facilitated the sharing of other artefacts, most notably code and miscellaneous documentation.

In general, formal project reporting for each investigated project took place on a weekly basis with the circulation of a project status report to relevant internal and external stakeholders. However, the SDM referred to the sizeable effort required in collating this information from a detailed project level up to management level as the "biggest priority" for forecasting, cash collection and deployment of resources. Furthermore, with many of the requirements being communicated by email, telephone and from multiple sources, project tracking frequently became unmanageable, leading P3-SE1 to say: "I'm finding that there are too many documents...They also brought in a process document at some stage. I don't know when this crept in." This problem was exacerbated by the dispersion of many practitioners across different projects and multiple sites and the lack of standardised document templates.

Another issue highlighted by PM1 related to the dearth of internal meetings with ABC management themselves. Describing it as "a gap, a break in communications," PM1 expressed acute frustration at the lack of face-to-face

interaction with ABC's management to discuss the project's difficulties as well as project planning, resource utilisation, and training needs. Also, both P2-SE1 and P2-SE2 complained that there were too few meetings where the whole project team got together to discuss the overall project strategy as well as difficulties experienced. In particular, both engineers made reference to five significant yet unplanned technical architecture changes insisted upon by the client without extension to the initial project deadline. In this case, PM2 was based in Dublin. This situation seems to contrast sharply with the plethora of meetings reported at a number of the client sites, leading PM1 to state, *"The guys [i.e., software engineers] are a lot more productive offsite because there are less distractions. With large companies and multi-nationals there tends to be a lot of meetings taking place and people coming up asking questions and basically taking away from their time."*

Communication delays were also reported. On P2, for instance, delays were incurred due to issues and requests having to percolate up the organisational hierarchy before being decided. In one instance, engineers had to wait seven work days before receiving sign-off on a proposed change to a shared piece of code. According to the SDM, these delays exerted additional and unnecessary pressure on often "overly aggressive" delivery schedules. Similar delays were reported on the other projects.

A number of steps have been taken to address the outlined difficulties. Issues pertaining to timely communication are raised in weekly status meetings and formally logged in an issue tracking spreadsheet. In many instances, however, it has been necessary to escalate the experienced delays to a higher level of management at the client site before being addressed. Whether these delays are intentional or not is difficult to ascertain. However, this appears to be a systemic problem experienced across most projects. As regards the content and format of information exchange, a generic communication plan is being developed that will serve as the basis for all project communication, both internally and externally. The success of this endeavour will largely depend upon the design of template documents which will capture essential information at the requisite level of detail as well as documenting the procedures that go with it. To achieve this, a number of best practice models are being taken into consideration, including the PMBOK [22]. The implementation and success of this plan, however, will depend on securing buy-in from ABC and client staff alike.

5.2 People Management

Activities underpinning DSD projects involve management of project "things" and people [3]. The former encompasses quantifiable aspects of a project, including requirements management, scope definition, estimating and scheduling, change management and integration efforts. In contrast, the people dimension includes the usually non-quantifiable characteristics of client satisfaction, vendor satisfaction, team morale and communication. Although many of the intricacies attributable to truly global separation did not apply in this study, as clients were based in

relatively close proximity to ABC, nonetheless, this study unveiled a number of issues pertaining to the people oriented aspects of DSD and outsourcing.

According to PM1, “the most crucial aspect of the whole thing [i.e., management of projects] seems to be managing people...internally and externally”. Elaborating on the management of external resources, PM1 described a situation where two contractors unexpectedly departed in the middle of project P1, resulting in a lot of upheaval and effort in locating suitably qualified replacements. This problem was not confined to P1 alone, however, as similar difficulties were experienced in filling technical lead and developer roles for the follow-on project also.

Others risks were also emphasised in relation to contractor dependence. P2-SE2 described how an incomplete and overly-optimistic specification was furnished by a contractor coming to the end of her contract, resulting in permanent staff working an excessive number of hours to complete the underlying work. In addition, many contractors held key roles, yet the transfer and sharing of domain and technical knowledge with ABC’s staff was described as “economical” by the SDM.

A lack of “teamness” also emerged from the data. P4-SE1 provided offsite assistance from ABC’s development centre to an internal team of engineers at client C4. However, despite spending an initial period of time onsite, P4-SE1 felt isolated from the team, citing exclusion from the team’s distribution list and meetings. P4-SE1 also expressed concern over the oscillating nature of the work delegated to her.

Similarly, both software engineers working on P2 pointed to the lack of unity experienced within the dispersed project team. P2-SE1 made reference to the inefficient exchange of information as a significant impediment to project progress. With specific reference to the extent of status reporting and ongoing meetings between team members across the sites, P2-SE1 remarked, “...developers were not allowed to move forward as a team. Instead they were treated as individuals and were often working in isolation for prolonged periods.” P2-SE2 corroborated this view, adding, “...you would not move collectively in the one direction, you would all be moving in different directions.” Others issues cited by these engineers included a reluctance by C2 to involve onsite ABC personnel in relevant meetings, failure to circulate project status, and hesitancy in communicating revised project plans. P2-SE2 summed up the situation as follows: “We are stuck in the middle and are being pulled on from both sides”.

It is clear that the issues outlined exist at an internal and an external level. At an internal level, management at ABC hope to phase out, over time, the dependence on contractors in preference for permanent staff. However, due to limited knowledge and expertise in certain technical and business areas, the degree to which this is implemented and success is achieved will largely depend on management’s commitment to provide sufficient levels of training to staff. The exact dispersion of team members is also being reviewed as many of the interviewees have expressed concerns at the physical displacement of team

members from each other. Furthermore, some engineers have expressed grievance at the different styles of management used by individual project managers, with variances being reported in terms of “micro” and “macro” management. The lack of consistent process is considered by many of those interviewed as a plausible reason for this problem.

At an external level, weekly status reports include a section detailing issues and delays incurred or threatening project success. Management at ABC strive to meet with peers at each client company on a two-monthly basis to discuss each project’s performance. However, as intimated by the SDM and PM1, a great degree of diplomacy is required in communicating and negotiating some of the hurdles impeding the relationship between ABC and its clients. Specifically, the inter-organisational aspect of outsourcing relationships ensures that there will inevitably be a “them” versus “us” mentality lurking somewhere beneath the surface. The degree to which this reveals itself will vary, but, with today’s increasing demands on maximising profit, and the proliferation of outsourcing as a vehicle to achieve this, it would be imprudent for client and vendor management alike not to accept that conflicting motives do exist. As projects undertaken by ABC are generally short-term and “rolling” in nature (three to six months in duration), management are only too aware of the consequences of not meeting the high expectations set by its clients. Failure to recognise this can jeopardise future projects with those clients. In the context of a small company such as ABC, this can have a detrimental impact on survival. On the other hand, care has to be taken not to subject employees to undue pressure. In the case of ABC, getting this balance right is proving extremely difficult to achieve.

5.3 Trust

Trust is considered a key component of all work relationships. Its significance is magnified, however, in the context of inter-organisational arrangements where participants are, for the most part, strangers to each other and frequently have competing goals. Although not as pronounced in this particular study, temporal and geographic distance has been shown to exacerbate issues relating to trust, where, for example, language barriers and cultural nuances come to the fore. In a context where project work is dependent on the harmonious co-existence of practitioners from different companies and backgrounds, establishing trust can be extremely difficult. In many instances, it is not uncommon for internal resistance to emerge as a significant inhibitor threatening the overall viability of such relationships. The existence of divergent goals and priorities frequently unite to undermine collaboration and cooperation activities. The data collected during this study reveals a number of instances supporting these concerns.

PM1 described a situation regarding the migration of permanent members of a team based at the client’s site back to ABC’s software development centre. Contractors comprising the team felt particularly threatened by this decision and felt unsure as to what long-term implications it would have for them. This action contributed to an air of suspicion pervading the work relationship between ABC and the contractors, culminating in a number of contractors not renewing their

contracts. According to the SDM, a number of contractors were “not really supportive of” the offsite dimension of ABC’s delivery model.

Another encounter highlighted with respect to distributed team members revolved around the indulgence of some team members in “finger pointing” and “blame-game” scenarios. According to PM1, this was especially true in situations where offsite members were not there to defend themselves, and even more so in the case of external contractors. For instance, intervention was required to resolve issues where onsite contractors attributed blame to remote team members. Interviewees working on P2 also reported inertia in terms of decision making and request turnaround time, prompting one of them to state: “They [client staff] definitely felt threatened by us... that we might take away all their work and leave them without a job.” Elaborating on this, P2-SE1 claimed client staff had a tendency to “keep their heads down” and didn’t want to assume responsibility for any decision making.

Summing up the difficulties experienced on project P2, P2-SE2 made the observation that “...due to the nature of the game... it is in their [i.e., client staff members] best interests not to cooperate with us. There is a clear conflict of interest.” However, he also argued that a large proportion of the difficulties encountered on project P1 had arisen due to “inadequate project management” on behalf of ABC. Overcoming trust issues is proving difficult to achieve. From an internal perspective, the establishment of a Sports and Social Club at ABC is considered a first step to ingraining better morale and rapport between all staff employed by ABC. Many concerns stem from the lack of long-term job security so priority is being given to the negotiation of longer-term commitments from clients. From the point of view of strengthening inter-organisational relationships, however, a clear solution is not so forthcoming.

5.4 Skills and Training

The process of developing software demands technical as well as domain competency. In a multi-client context, the range of these skills can vary enormously depending on the nature of the work and the area of specialisation of the client company. Harnessing the fundamental skills and knowledge required for a variety of business domains is a challenging undertaking. Furthermore, the volatility of the overall business landscape coupled with technological advancement demands a great degree of flexibility, versatility and commitment to continually hone and foster skill sets. In small company settings, where budgets are inevitably tight, such challenges assume added resonance. Inter-organisational work contexts exacerbate these difficulties even further, inevitably expediting the need for continual adaptation and retraining. Data collected from ABC revealed a number of issues pertaining to skill and training levels.

Some interviewees perceived that their technical skills were becoming outdated, particularly those involved in mainframe-based projects. Apprehension and frustration was expressed by a number of interviewees assigned to projects without, in their opinions, a requisite level of training and expertise to meet the

demands of their particular jobs. At management level, for instance, PM1 emphasised the lack of formal project management training provided for the role. Similar reservations were conveyed by practitioners across other projects. Engineers working on project P2 expressed dismay at the brevity of training provided for a new technology: “We were given half a day’s training and then had to pick it up as we went along”. P4-SE1 had to conceal a lack of experience in a technical area from the client while covertly developing those skills through on-the-job training. Lack of training contributed to additional unscheduled effort being devoured on time set aside for project-specific tasks, leading to mounting pressure and frustration being absorbed by frontline staff.

Furthermore, this lack of training manifested itself in a fear of being exposed in dealings with clients. The study suggests that this is particularly relevant in client settings, where interviewees reported greater levels of observation by client staff and management alike. P4-SE1 claimed that by being onsite there was a feeling that “someone is keeping an eye on you all the time...waiting for you to make a mistake and show you up”. PM1 concurred with this observation, expressing trepidation towards staff being “thrown in at the deep end,” thus mitigating the likelihood of project success. As a short-term measure to offset the lack of internal technical expertise available in certain technologies and programming languages, management at ABC recruited a number of software contractors. While the SDM acknowledged that many of these contractors possessed specialised expertise and held key positions, he stressed that this was a “temporary measure” and that internal staff were being groomed to fill those positions on future projects.

To overcome many of the burdens experienced by its practitioners, more formal measures are beginning to be introduced by ABC’s Human Resource department. Specifically, quarterly performance reviews are being scheduled which will formally map out and track individual performance goals and objectives for each employee and identify training needs. These training needs will then feed into an overall training plan which management intends to implement using internal and/or external training courses.

5.5 Process

Proponents of software process argue that production will most likely descend into chaos without the support structures it provides, and that improved software process will improve software quality. Process is described as “the glue that ties together people, technology, organisational structures and management in a coherent whole, focusing on the business goals and objectives” [23]. However, sound process on its own does not guarantee a good product or end result - social and technical factors have equally critical roles to play [24].

ABC inherited a project management process based primarily on the PRINCE2 methodology from its preceding company. This process provides a detailed description of activities to be performed by project managers while also including templates to be used for tasks such as issue tracking and status reporting. In general, all proposals for new business make reference to this process. However,

for one reason or another, virtually all interviewees participating in this study were either unaware of the existence of this process or simply did not follow it. While acknowledging that this process was an integral part of any project proposal to its clients, PM2 added that once a project commences, no explicit reference is made to it. Projects managed by PM1 also did not take account of this process. SDM admitted the use of this process model was not mandatory as, ultimately, individual client requirements would dictate what, how and when certain activities took place.

In this study, C1 were the only client company with formal software process accreditation, having been formally assessed at Level 3 against the SEI's Capability Maturity Model Integration (CMMI) model. At the behest of C1, all ABC's team members involved in project P1 underwent introductory training in CMMI at the client's site. Aside from providing an overview of the Key Process Areas within the CMMI model, this training also incorporated policies, procedures and documentation to be used and complied with during the course of the project. Despite adhering to the practices and activities of the model, a number of difficulties were encountered. For instance, there was a discrepancy of 40% in estimates generated using the method recommended by the client. This resulted in the client's management exerting more rigorous project monitoring and control on subsequent project work. Also, despite being considered good practice, there was no formal requirement for conducting post-project reviews, thus increasing the likelihood of issues reoccurring on future projects. PM1 strongly felt the existence of a formal project management process within ABC would bring greater consistency, transparency, accountability and ownership to the management of software projects. Similarly, PM1 believed the absence of such a process was a "key thing", the "missing link", and "probably the reason why there is no buy-in from management".

In contrast to C1, no formal process model was implemented by C2. The data collected identified the lack of process as a significant difficulty affecting the success of the project. The SDM commented that the "greatest concern...[was caused by] a lack of formality and structure" in projects undertaken with C2. Indeed, the SDM believed that ABC's willingness to operate within such a "totally chaotic" environment was a significant factor in ABC being chosen over larger and more illustrious competitors. Furthermore, on projects P2 and P3 it seems project estimates were largely determined by ABC's project manager with little input from the developers assigned to them. While it was too early to assess the impact of this on P3, developers on P2 expressed concern and resentment at this practice as estimates were essentially adjusted to fix rigid schedules imposed by the client rather than by those implementing the project. Invariably, this amplified the pressure on an already risky project, resulting in staff enduring many late nights and weekends clawing back unscheduled project effort.

The Test Lead working on P3 also lamented the lack of process to govern project development. Relying on past experience, P3-SE1 expressed frustration that there were no explicit guidelines available to indicate how the project should proceed. It appears C3 themselves had no standard process in place to facilitate project

development, thus necessitating P3-SE1 to conduct searches on the internet for suitable template documents relating to testing along with guidance on how best to approach it.

In general, the data collected suggests that software development represented a trade-off between time-to-market and the quality of the end product. As a result, quality-related activities, including peer reviews and code walkthroughs, were frequently sacrificed on many of the projects in order to satisfy rigid deadlines. In attempting to demystify the reasons for why these activities were compromised, both P2-SE1 and P2-SE2 point to over-eagerness to secure more long-term work with the client, arguing the estimation of effort and time often did not make sufficient allowances for recognised quality-adding practices.

To alleviate the many difficulties experienced in the delivery of their software projects, management at ABC embarked on a software process improvement (SPI) initiative. As a first step, ABC requisitioned a gap analysis between the company's own internal process and the Software Project Planning Key Process Area within the CMMI. Discrepancies between both models were reconciled into ABC's own existing process model. Despite this, however, the early momentum and enthusiasm demonstrated at the outset of the initiative were not sustained. Instead, client projects took precedence over training and the rollout of the updated process model across the company. This inertia coincided with the sudden and premature departure of the previous SDM. However, following a few months of inactivity, ABC embarked on a subsequent SPI initiative. In particular, a SCAMPI Type C assessment was conducted and a new Process Improvement Consultant role was incorporated into the company. The results of this assessment are currently being analysed by the PIC with action items being formulated as part of an overall process improvement plan. ABC intends to conduct a more formal assessment at a future date.

6.0 Conclusions

The practice of developing software is continuously evolving from a centralised, co-located model towards a more distributed, virtual version. This evolution is happening both at an inter- and intra-organisational level globally. This study investigated the experiences of a small, Irish software provider involved in inter-organisational software outsourcing with a number of the clients companies based in the same jurisdiction. Although the context is somewhat diluted in terms of direct application to a truly globally distributed scenario, nonetheless, several parallels can be drawn with issues highlighted in extant literature and devoted to a globally distributed landscape, including that which involves larger organisations. In particular, this study articulates difficulties experienced in relation to communication, people management, trust, skills and training, and process in the development of software, and briefly outlines how the company is dealing with these challenges. The study highlights financial and resourcing constraints, particularly in relation to training needs. Also, the cost of implementing a full-blown quality/process model, such as the CMMI, was seen as too debilitating and

not totally applicable to the work being undertaken at ABC at this moment in time. Consequently, the company is investigating other more agile models and approaches, with cost and effort inevitably being the overriding concern.

Although specifically capturing data from the perspective of the software provider, the findings of this study are equally of interest to client organisations engaging in outsourcing relationships with small software providers. Future work will chart ABC as it expands its software development operations into the U.K. market. By introducing an international dimension to this study, it is envisaged that many of the nuances reported here will be further accentuated, with additional issues also possibly emerging. The authors propose research from the viewpoint of client organisations soliciting the services of small vendors in a DSD context. Only by incorporating the viewpoints of both sides can the magnitude of the complexities underpinning these relationships be fully understood, and hence addressed.

7.0 Acknowledgements

This research was funded by the Science Foundation Ireland through the Global Software Development for SMEs project at the University of Limerick.

8.0 References

- 1 Carmel E, *Global Software Teams: Collaborating Across Borders and Time Zones*. Prentice-Hall 1999, ISBN 0-139-24218-X
- 2 Karolak D W, *Global Software Development: Managing Virtual Teams and Environments*. Wiley-IEEE Computer Society 1998, ISBN 0-818-68701-0
- 3 Rad P F and Levin G, *Achieving Project Management Success using Virtual Teams*. J. Ross Publishing 2003, ISBN 1-932-15903-7
- 4 Ebert C and De Neve P (2001). Surviving global software development, *Ieee Software*. 18, 62-69.
- 5 Herbsleb J D and Grinter R E, Splitting the Organisation and Integrating the Code: Conway's Law Revisited, Proceedings of the 21st international conference on Software engineering 1999, pp85-95, LA, CA, USA, 1999
- 6 Prikladnicki R, Audy J L N, and Evaristo R (2003). Global Software Development in Practice: Lessons Learned, *Software Process Improvement and Practice*. 8, 267-281.
- 7 Dybå T, Factors of Software Process Improvement Success in Small and Large Organizations: An Empirical Study in the Scandinavian Context, Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering 2003, pp148-157, Helsinki, Finland, 2003
- 8 Hatten T S, *Small Business - Entrepreneurship and Beyond*. Prentice Hall 1997, ISBN 0-131-80340-9
- 9 *European Commission: Observatory of European SMEs 2003 (No.7) - SMEs in Europe 2003*,

- http://europa.eu.int/comm/enterprise/enterprise_policy/analysis/doc/smes_observatory_2003_report7_en.pdf (visited February 2006)
- 10 Casey V and Richardson I, Virtual Software Teams: Overcoming the Obstacles, 3rd World Congress on Software Quality 2005, pp63-70, Munich, Germany, 2005
 - 11 Herbsleb J D, Paulish D J, Bass M, Global Software Development at Siemens: Experience from Nine Projects, Proceedings of the 27th International Conference on Software Engineering 2005, pp524-533, St. Louis, MO, USA, 2005
 - 12 Hofstede G, *Culture's Consequences: International Differences in Work-Related Values*. Sage Publications 1980, ISBN 0-803-91306-0
 - 13 Olson J S and Olson G M (2003). Culture Surprises in Remote Software Development Teams, *ACM Queue*. 1, 52-59.
 - 14 Sparrow E A, *A Guide to Global Sourcing: Offshore Outsourcing and other Global Delivery Models*. BCS 2005, ISBN 1-902-50561-1
 - 15 Willcocks L P and Lacity M C, *Strategic Sourcing of Information Systems: Perspectives and Practices*. John Wiley and Sons Ltd 1998, ISBN 0-471-97787-X
 - 16 Sabherwal R (2003). The Evolution of Coordination in Outsourced Software Development Projects: A Comparison of Client and Vendor Perspectives, *Information and Organisation*. 13, 153-202.
 - 17 Yin R K, *Case Study Research: Design and Methods*. 3rd ed. Sage Publications 2003, ISBN 0-761-92553-8
 - 18 Bryman A and Bell E, *Business Research Methods*. Oxford University Press 2003, ISBN 0-199-25938-0
 - 19 Lubbe S (2003). The Development of a Case Study Methodology in the Information Technology(IT) Field: A Step by Step Approach, *Ubiquity*. 4, 2-2.
 - 20 Marshall C and Rossman G B, *Designing Qualitative Research*. 3rd ed. Thousand Oaks: Sage Publications 1999, ISBN 0-761-91340-8
 - 21 Miles M and Huberman M, *Qualitative Data Analysis: An Expanded Sourcebook*. 2nd ed. Thousand Oaks: Sage Publications 1994, ISBN 0-803-95540-5
 - 22 PMI, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 3rd ed. Project Management Institute 2004, ISBN 1-930-69945-X
 - 23 Zahran S, *Software Process Improvement - Practical Guidelines for Business Success*. Addison-Wesley 1998, ISBN 0-201-17782-X
 - 24 van Solingen R (2004). Measuring the ROI of software process improvement, *Ieee Software*. 21, 32-38.