Virtual workgroups in offshore systems development

Sachidanandam Sakthivel*

Department of Accounting and MIS, College of Business Administration, Bowling Green State University, Bowling Green, OH 43403, USA

Received 22 March 2004; revised 27 August 2004; accepted 6 September 2004
Available online 14 November 2004

Abstract

The market for offshore systems development, motivated by lower costs in developing countries, is expected to increase and reach about $15 billion in the year 2007. Virtual workgroups supported by computer and communication technologies enable offshore systems development. This article discusses the limitations of using virtual work in offshore systems development, and describes development processes and management procedures amenable to virtual work in offshore development projects. It also describes a framework to use virtual work selectively, while offshore developing various types of information systems.

Keywords: Offshore systems development; Global software development; Global outsourcing; Virtual work

1. Introduction

Offshore Systems Development (OSD) involves development of information systems in another country. The Meta Group, Inc. estimates the OSD market at $7 billion today and expects it to grow more than 20% annually; it also estimates the market at $10 billion by the year 2005 and $15 billion by the year 2007 [103]. The Wall Street Journal reported that the estimated export of software and technology services from India alone is about $12 billion during the year 2004 [99]. A number of recently published books and articles [46,92,97,98,99] show the growing interest in OSD among practitioners and researchers. Although many advantages are cited in literature, the primary motivation for OSD is the low cost of systems development in developing countries such as India, Russia, China, and Bulgaria [29,39,107].

Advances in communication technologies and virtual work using these technologies facilitate OSD. Virtual workgroups consist of geographically dispersed people working interdependently with shared purpose across space, time, and organizational boundaries and using technology to communicate and collaborate. Virtual work enables an organization to combine the best expertise at lower costs despite geographic separation.

Virtual workgroups can use technologies ranging from telephones to specialized multimedia software products designed for a specific group task [46,100]. The technology in virtual workgroups depends on whether the group members interact synchronously or asynchronously. Face-to-face (F2F) interactions offer the richest synchronous communication, but virtual group members separated by distance may use rich synchronous communication media such as television video conferencing and less-rich synchronous communication media such as desktop video conferencing as a substitute for F2F interactions. Software products such as NetMeeting enable real-time collaborative work. Customized collaborative work products may include group support systems, CASE tools and shared application development data repository and run on the Internet, on the company’s virtual private network, or on its wide area network. Several application service providers offer products and services with secure facilities for virtual work over the Internet [108]. These products allow desktop video conferencing, audio conferencing, chat facilities, white boarding, file sharing, application sharing, and remote control of computing resources. Virtual groups may also use lean synchronous communication media such as telephones, conference-calls using telephones, conference-calls on the Internet, instant messaging systems, and chat facilities.
Group members, who are interacting asynchronously, may use emails, file transfers and voice mails. Table 1 summarizes various communication media used in group work.

Virtual work in OSD involves people from various countries and cultures speaking various languages, and separated by time and distance. Since cost reduction is the primary motive for OSD, virtual work is necessary to avoid travel to offshore countries and to facilitate collaborative work globally. Although several books and articles discuss the use of virtual work across the boundaries and argue that the use of communication technologies can alleviate problems caused by separation of workers in time and distance [17,46,60], a study of a global virtual group performing concurrent engineering concluded that virtual groups are not as effective as they are publicized [9]. In addition, global virtual groups do not create the expected value [26]. Research studies in psychology, organizational behavior, management, and communication show that global virtual work has certain limitations. To achieve cost savings without sacrificing quality and time, OSD projects should employ virtual work selectively. This article describes systems development processes and management procedures amenable to virtual work in OSD projects. It also describes a framework to use virtual work selectively while offshore developing various types of information systems.

Section 2 shows that numerous and complex interactions among group members reduce the effectiveness of virtual work in knowledge intensive collaborative tasks, and defines task coupling. Section 3 shows that diversity among group members also reduces the effectiveness of virtual work, and defines group cohesion. Section 4 defines virtuality levels that vary with task coupling and group cohesion, and discusses how virtual group members should interact depending upon the virtuality level. Section 5 discusses virtuality levels associated with various development processes, and identifies development activities amenable to virtual work in OSD. Section 6 discusses virtuality levels associated with various project management procedures, and identifies procedures amenable to virtual work in OSD. Section 7 discusses virtuality levels of development activities for various types of information systems, and identifies those development activities suitable for virtual work consistent with the cost saving objective of OSD. Lastly, this article suggests future research directions and concludes the discussion.

2. Task coupling in virtual work

Studies show that systems development involves individual cognitive activities that need uninterrupted periods to concentrate, and collaborative activities that require periods of interaction with others [84,85]. Research has established the importance of F2F interactions in collaborative activities. A major ethnographic study involving programmers separated by distance concluded that F2F interactions are necessary to convey complex information, and to build and sustain trust that is crucial in group work [79]. Co-presence and proximity support sustained creativity and knowledge-intensive work [58,84,85] that are characteristics of systems development. When tasks are unclear and need a consensus, co-presence and F2F interactions are essential [21]. Any innovative or creative work that involves partnerships, needs F2F interactions [19].

A study underscores the importance of co-presence, proximity, and F2F interactions with a finding that suppliers have difficulty innovating to meet new customer requirements when far from the market [104]. The study stresses the importance of shared context for creative work that electronic communication cannot adequately compensate. F2F interactions provide social context, shared experiences, nonverbal nuances, and contextual cues [94,95]. They also provide a shared context among group members, enable rich

<table>
<thead>
<tr>
<th>Real time or different time</th>
<th>Same place or different place</th>
<th>Type of communication</th>
<th>Communication media</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real time</td>
<td>Same place</td>
<td>Richest synchronous</td>
<td>F2F interactions</td>
</tr>
<tr>
<td></td>
<td>Different place</td>
<td>Rich synchronous</td>
<td>Television video conferencing, aided by collaborative work products that may include group support systems, CASE tools and shared application development data repository</td>
</tr>
<tr>
<td></td>
<td>Different place</td>
<td>Less-rich synchronous</td>
<td>Desktop video and audio conferencing, aided by collaborative work products that may include group support systems, CASE tools, shared application development data repository, chat facilities, white boarding, file sharing, application sharing, and remote control of computing resources</td>
</tr>
<tr>
<td>Real time</td>
<td>Different place</td>
<td>Lean synchronous</td>
<td>Telephones, conference calls using telephones, conference calls on the Internet, instant messaging systems, and chat facilities, and collaborative work products that may include group support systems, CASE tools and shared application development data repository</td>
</tr>
<tr>
<td>Different time</td>
<td>Same or different place</td>
<td>Asynchronous</td>
<td>Emails, file transfers, voice mails, and collaborative work products that may include group support systems, CASE tools and shared application development data repository</td>
</tr>
</tbody>
</table>

Table 1

Communication media in group work
communication that fully uses the physical senses and psycho-emotional reactions, and have capacity for interruption, repair, feedback, and learning [73]. Studies of systems development projects found that successful problem solving needs the development group members to have a common understanding of the work and a strong sense of identities provided by F2F interactions [3,82,96,105]. F2F interactions build shared understanding and definition of the task [69]. Whereas F2F interactions create an environment of common understanding, mutual respect, and emotional closeness that support the free expression and discussion of ideas, physical distance permits disinterest, and a wide variation in attitude and feelings [32]. In addition, discussions and interactions in virtual groups can overwhelm the group members, and reduce comprehension and understanding of the group work [15].

F2F interactions enhance the ability of group members to work across spatial and cultural divides. Creating a synergy is often difficult in virtual workgroups as compared with F2F interactions [54]. Virtual workgroups gain a negative synergy, and lose a positive synergy that F2F interactions can achieve [18]. Due to feelings of isolation and detachment, people in virtual workgroups are likely to be less productive and less satisfied compared with people working in an F2F environment [18,54,55]. Frequent F2F interactions are a key to success in virtual work [89].

In summary, knowledge-intensive collaborative tasks need co-presence, proximity, and F2F interactions of the group members in varying degrees for effective group work. Collaborative tasks that need creativity, innovation, and consensus also need co-presence, proximity, and F2F interactions. In addition, collaborative tasks require co-presence, proximity, and F2F interactions when these tasks have undefined or unclear objectives, involve problem solving, or need mutual understanding of group members. We define such tasks as having a high degree of coupling because these tasks involve numerous and complex interactions among group members.

3. Group cohesion in virtual work

Members in a virtual workgroup need strong identities to the project and to one another. However, these traits are less pronounced when group members are from different cultures and speak different languages [9]. Cultural differences increase the need for better communication and coordination in a virtual group [47]. A study of a global virtual group found that culturally disparate members create conflicts in virtual groups [35]. The study also found certain peoples work independently but others, who are used to working collectively, take longer time to adapt and have difficulties in adjusting to frequent shifts in workgroup and group membership. These people also believe in hierarchy and have difficulties in working with new leaders, and have difficulties in taking leadership positions in fluid virtual groups. The cultural differences have negative effects on formation of workgroups for projects of short durations and negatively affect the effectiveness of virtual work. Loose partnerships produce conflicts and hamper innovation [19]. Culturally diverse workgroups experience more conflicts, higher turnover, less trust, less job satisfaction, more stress, more absenteeism, and more communication problems [5,80,102]. Culturally diverse workgroups experience communication difficulties and interpersonal conflicts initially that may become less pronounced with F2F and longer synchronous interactions [8]. OSD may alienate in-house staff, trigger emotional and politicized debates, and create morale issues that affect both onshore and offshore projects [57].

Managing a global virtual team is more complex than managing a local virtual team [67]. Virtual workers, who rely on their virtual coworkers to complete their tasks experience time pressure, loss of control and a decline in personal productivity [25]. When the group is ambiguous about the goals and objectives, task conflicts (what should be done), process conflicts (how should it be done), and relation conflicts are created affecting the group performance [43]. Work independence, trust, and a sense of belonging are essential for success of virtual group work [87]. Trust among peers in a virtual setting is essential for effective communication, collaboration, and mutually acceptable ways of coordinating work [42,55].

The virtual work in OSD involves people speaking various languages, and, who come from various companies, and countries. Such diverse workgroups experience difficulties in communication and conflicts that negatively affect the effectiveness of virtual work. These groups may face difficulties in building relationships, shared views and trust that are important for effective virtual work. We define such groups as having low cohesion. A study identified the above factors that lead to low group cohesion as the causes of failure in global software development projects [48]. Besides the language and culture, different backgrounds, experience, skills, role perceptions, application domain knowledge, IT skills, and mental schema of group members reduce the group cohesion.

4. Virtuality levels and interaction media

Virtual work is often managed as if it is F2F work, which leads to failures. A major study of a global virtual group has many recommendations for an effective virtual group process [65]. The effectiveness depends on a fit between the message complexity in the group process and the form of interaction medium. Increased task interdependence in a group process will increase the interaction frequency, and increased task complexity will increase the message complexity, which in turn, affects the choice of the interaction medium. For example, to build commitment and trust, frequent interactions of group members with
a high level of involvement, and exchange of complex messages are necessary. Tasks with high couplings and tasks to improve group cohesion need F2F interactions over a long time. Virtual groups with low cohesion need F2F interactions and rich synchronous media to build trust and relationships, and share views [65]. As the team cohesion increases, the virtual groups can use less-rich, lean synchronous, and asynchronous communication media. Since group members, who are observably different in culture, language, or profession experience exclusions in communication, F2F interactions are necessary to build relationships and trust, and to reverse communication breakdowns [8].

Periodic F2F meetings of group members are necessary to successful development of a virtual group [93]. These F2F interactions and coordination meetings should be scheduled around challenging and complex tasks [65]. Communication in virtual work includes conveyance that involves sharing of information, opinions, and disagreements, and convergence that involves a critical examination of others’ views to reach a consensus. Planned and managed interaction behaviors that reduce the conveyance time in virtual work to about the time required in F2F interaction would increase the time needed for convergence, and thus increase the performance of virtual work [63].

Desktop videoconferencing provides a cost-effective substitute for F2F interactions and a means for meaningful interactions in a virtual workgroup environment. However, a study involving desktop videoconferencing found that group members, who are trained in its use and believed that it has benefits, had high user satisfaction with the system, but the group performance was low [101]. The study argued that universal and common desktop videoconferencing systems are not suitable for all types of tasks and people, and not all individuals will be successful in virtual groups. It added that virtual work needs training, exploration, and adaptation to personal work style to have an acceptable group performance. Coordination of virtual work is expensive, and virtual work is not effective for all types of tasks. Latest technological advances in videoconferencing using televisions may provide a good substitute for F2F interactions better than what was feasible about a decade ago, but it carries a risk of not saving costs while achieving acceptable performance in OSD. Since the investments in high quality videoconferencing systems, groupware, and bandwidth to support sustained real-time interactions are high, global virtual workgroups often use less-rich synchronous, lean synchronous, and asynchronous communication media. In addition, extended time zone differences (8 hours or more) between onshore and offshore countries create difficulties in synchronous communication by requiring either party to work at odd hours.

Asynchronous communication using computer-mediated technologies in virtual work needs more time and effort, and effective processes to manage the virtual interactions because asynchronous communication changes the sequence and synchrony of messages, and increases ambiguity [68]. Computer-mediated asynchronous communication is suitable for structured and well-defined tasks [64,65] such as coding according to system specifications. Such virtual works need early F2F interactions to define the project and its objectives, define the roles of members [88] and to enhance the effectiveness of later computer-mediated communications [56].

Groups that have low cohesion often run into interpersonal, task, or process conflicts, which need F2F interactions, or at the least, rich synchronous communication media to resolve them [70]. Asynchronous communication media create temporal patterns that are unsuitable for managing conflicts arising out of low group cohesion. In addition, the time required both to convey disagreements and to converge on a resolution would increase and affect the group performance. Methods of managing conflicts in an F2F environment are not transferable to a virtual group working asynchronously. Asynchronous media such as email are unsuitable for managing conflicts in virtual workgroups.

The OSD involves some individual work and some group work of diverse members separated by time and distance. Group work that has a high degree of task coupling for a group with a low degree of cohesion should be done in an F2F environment. Such a group work has a very low virtuality level. Group work that has a low degree of task coupling for a group with a high degree of cohesion can be virtually done, employing lean synchronous and asynchronous communication media. Such a group work has a high virtuality level. User background, skills, knowledge, and training in virtual work should guide the selection of the appropriate technology [9]. Table 2 shows varying degrees of task coupling and group cohesion that define virtuality levels and their corresponding interaction media.

5. Systems development processes in virtual projects

Systems development approaches can be classified into structured, iterative and incremental, and integrated approaches. The selection of the right development approach is necessary to address issues such as lack of users’ knowledge of application, lack of users’ involvement, missing, incorrect, and evolving requirements, and risks associated with new technology and poor quality [14]. Using an inappropriate development approach has the risk of increasing development time and cost, and in jeopardizing quality while increasing the task coupling. Virtual work with an inappropriate development approach would further increase the task coupling and risks.

Structured approaches. The number of methodologies in structured development approach exceeds 100 [74–78], and includes the popular methodologies of Yourdon and Constantine, DeMarco, and Gane and Sarson. Most methodologies in this class follow the waterfall model.
In structured development approaches following the waterfall model, the development of systems in stages provides modularity to development, and enables individuals and small groups to work independently. However, according to Brooks, software development, an exercise in complex relationships, needs great efforts in communication among involved parties that diminish the benefits of modularity [16]. The task coupling in the waterfall model depends on the development stage. The task coupling in the requirements analysis stage is high because it requires major user participation in organization analysis and business process design to meet user needs. The highest degree of interaction occurs between the user and the analyst during the requirements analysis stage [59]. Regardless of the methods and tools employed, the success of requirements analysis depends on how well the users and analysts communicate [40]. A study showed that most requirements are difficult to identify, and identified requirements are unclear and not well organized [24]. The study also showed that these problems are exacerbated in global software development. Since the cost of detecting and correcting errors at a later stage is high, quality assurance procedures should detect and correct requirement errors at an early stage. Initial studies show that distributed quality assurance of software code is feasible [13,37], but distributed quality assurance of requirement definitions that need high user involvement would have a high task coupling until requirement definitions are formalized and made user-friendly. Requirements analysis methods, even in colocated places, are not very effective, and therefore, need more research [61]. Additional research is needed to have effective requirements analysis methods and tools in the global development environment [22,23,86,109].

Among the design activities, the user interface design needs an iterative group process with high user involvement, and therefore, involves high task coupling. However, with a user interface prototype, shared application development data repository, group support systems, and rich or less-rich synchronous communication, this task coupling can be reduced to moderate level. Most other design activities, coding, and testing have low task couplings, provided requirements and design do not change frequently. Because the system implementation may need quick resolution of unforeseen problems, it is an on-site activity. The maintenance stage that predominantly involves coding and testing with clear specifications also has a low task coupling. Overall, development activities that use well defined, well documented and frozen requirements, and that do not require much of the collaboration with users have low task couplings.

Iterative and incremental development approaches. Iterative and incremental approaches include rapid prototyping, rapid application development, and agile development methods.

Rapid prototyping, also known as quick and dirty development, has many variations [72]. In rapid prototyping, a system is quickly coded after a short requirements analysis and design phase. The developed prototype then goes through several trials with users and modifications, before it is implemented and documented. Rapid prototyping is less time-consuming and less expensive, when the required system is simple, small (6 person-months), or has short life (throw away systems). Rapid prototyping is also useful to test the technical feasibility before a full-fledged system is developed or to provide proof of concept. The initial and short requirements analysis and design phase has a high task coupling, because it needs heavy user involvement as in the requirements analysis stage of the waterfall model. As rapid prototyping is used for small systems, the number of users would be small. The task
coupling arising out of user interactions in multiple trials and user feedback can be reduced to moderate level with a prototype, shared application development data repository, less-rich synchronous communication, and collaborative work products.

Rapid Application Development (RAD) has several variations such as Martin’s [62] and McConnell’s [66] approaches. In general, it has requirements planning, design, construction, and cutover phases [52]. The RAD uses CASE tools, group support systems, and time boxing concepts to control development time. The requirements planning phase often includes joint application development exercises that involve many participants, and has a high task coupling. Since RAD calls for experienced users, analysts, and programmers to develop a system quickly, the development exercise is often intense with high task coupling. During the iterative design and construction phases, the conveyance of user feedback with the help of a partially developed software product, a CASE tool and group support systems involves low task coupling, but convergent issues that need resolution have a high task coupling. Whereas the design part would have a moderate to high task coupling, the construction part would have a low to medium task coupling. The cutover stage, with concurrent testing, training, and system conversion involving users, has high task coupling. The maintenance using RAD is very similar to development on a smaller scale.

Object-oriented development (OOD) also has several variations that include the approaches of Booch, Coad and Yourdon, Jacobson, and Rumbaugh and use a tool suite such as Rational Rose. Although OOD has analysis, design, and programming phases (workflows), these are integrated and unlike the waterfall model. The OOD often employs an iterative and incremental development approach like RAD.

The Agile Development Methods (ADM) include Scrum, Dynamic Systems Development Method, Crystal Methods, Feature-Driven Development, Lean Development, Extreme Programming, and Adaptive Software Development [38]. These methods are generally incremental, co-operative, and adaptive [1]. The ADMs emphasize individual interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to changes over following a plan [4]. These methods cover different/certain phases of the software life cycle and lack proper project management support. In addition, these methods suggest universal solutions to various development situations and may not be suitable for large systems [2]. The ADMs need intensive and continuous participation of users and daily interactions, often F2F with developers. The high task coupling at all stages of ADMs requires excellent communication and problem resolution methods. Although ADMs have low task coupling for the conveyance of user feedback, they have high task coupling for convergent issues that need resolution. While developing small systems, ADMs can use partially developed software products, shared application development data repository, and group support systems to reduce the task coupling to moderate level.

**Integrated approaches.** The integrated approaches include incorporation of prototyping within the waterfall model, and spiral model approaches. The spiral model involves planning, risk analysis, prototyping, verification, and validation at each stage of development. These activities during the proof of concept and requirements analysis have high task couplings. In addition, the prototyping during each stage has high task coupling. The user interface design, as in structured approaches, has moderate task coupling. Other design, coding, and testing stages have low task couplings. The implementation stage has a high task coupling. The use of spiral model in development of business information systems is uncommon because it is more time-consuming and expensive than structured approaches. Nevertheless, offshore development of those stages with low task couplings can make it less expensive.

Since unclear and changing requirements are common with many systems, the waterfall model can be complemented by prototyping to elicit, verify, and validate requirements. The task couplings in this combination approach are as with the waterfall model that has stable requirements.

**Methods, tools, and procedures.** The quality of systems development methods, tools, and procedures within a development approach determines the quality of a system [14,27,50,83]. It includes methods, tools, and procedures for quality assurance. Virtual work with different or unsynchronized methods, tools, and procedures in dispersed locations increases the coordination and communication efforts, and the task coupling.

Group members working virtually across time and space involve in collaborative activities such as problem definition, problem solving, knowledge acquisition, and knowledge transfer. Such virtual work needs to translate into material production of code and implementation. The material work is an extension of virtual work and has an electronically mediated representation in the form of various documents. Correct intertwining of material work and virtual work is necessary to reinforce, complement, synergize, and reciprocate the work of group members. In instances, where one or more of these aspects are lacking or inadequate, unexpected negative outcomes can emerge and affect the group performance [90]. The importance of documentation in systems development is well established. The OSD requires synchronizing the documentation in dispersed locations such that work proceeds from one place of activity to the next, and it passes in timely and coherent fashion. In addition, it requires correct implementation of plans and procedures associated with development. Implementation of such communication and management systems across dispersed locations is difficult. Ethnographic research suggests that virtual work in dispersed locations creates a ‘buck passing’ culture and depends on excellent communication for success [41]. The lack of reference
points for coordinating the flow of work by time, place, or sequence can lead to disjointed interactions [81]. A successful temporal coordination mechanism that influences these interaction behaviors leads to better performance of virtual workgroups [63]. Without adequate intertwining of material and virtual work, and without management systems to coordinate and document development activities in dispersed locations, task couplings for development activities in virtual projects will increase, and virtuality levels will decrease. The iterative and incremental approaches that place less emphasis on documentation, and depend on continuous user interactions to aid the development process may use partially developed products as reference points for conveyance, but issues that need convergence would have high task coupling. Lack of group cohesion would further decrease the virtuality level.

Although structured approaches that follow the waterfall model provide modularity to development and facilitate virtual work in certain stages, unclear and changing requirements with heavy user involvement create high task couplings. The iterative and incremental development approaches help to manage changing requirements, but most activities other than construction have high task couplings. New development approaches such as ADMs are designed for an F2F environment, and are not proven for large systems. To control risks and develop high quality systems iteratively, the spiral model is useful, but it is time-consuming and expensive. The combination of prototyping and the waterfall model helps manage unclear and changing requirements to some extent and provides modularity in subsequent stages.

6. Management of virtual projects

The research literature is replete with scores of study on managing projects that use structured development approaches. Estimating the development effort has been an intractable problem in project management [30,44]. Currently available tools may not provide an accurate estimation of development efforts in a virtual environment. Since some stages of development may take place in an F2F environment and other stages virtually, poor estimation of effort involved in each stage will lead to inaccurate project schedule and resource allocation in each type of work. Underestimated IS projects may sacrifice the quality while pressuring the project to bring under estimates [45]. Underestimations may disproportionately reduce the time needed to share information, opinions, and disagreements, and the time needed to examine others’ views to reach a consensus in virtual work, and further jeopardize the quality. Geographic separation of group members adds difficulties to setting goals, budgets, schedules, and resources [67]. Without accurate estimations, realistic schedules, and appropriate resource allocations, the task coupling in management of virtual projects would be high.

Project management methods and tools vary depending on the nature of the project, and no single approach is universally applicable to all types of projects [7]. For example, strategic systems and systems with unclear requirements need management by a steering committee consisting of users. The user manager should lead the project, resolve conflicts, and review progress periodically. In OSD, such project management activities have a high task coupling. Although formal planning and control tools such as PERT are useful, they are inadequate. The maintenance of informal communication networks, not possible in virtual work, is an important determinant of project success [53]. In contrast, the IT group can use formal planning and control tools to manage non-strategic applications with stable requirements.

Virtual group members should have not only the skills, expertise, and high productivity in system development, but also need skills and training in virtual work and various virtual work technologies [100]. IT people often implement virtual group projects as IT projects without considerations to training, communication, human, process and management issues in a virtual environment. Virtual work proponents often stress the importance of technical skills and ignore the importance of interpersonal skills [100]. Virtual group members need training both in the virtual technologies and in the virtual interaction processes. People, who are better disposed to virtual work in contrast to independent work, succeed as virtual group members.

Project management challenges in global virtual work are higher than the challenges in an F2F environment [67]. Virtual work is often managed as if it is F2F work, which adds to failures. The management of virtual group members should be different from the management of workers in an F2F environment. The project manager has a greater role to coordinate, communicate, facilitate the group process, and ensure participation of all members [9]. Besides recognizing the individual contribution in virtual work, the project manager should build an appropriate reward structure, as an inappropriate reward structure may kill motivation and contribution. The lack of referents—the ‘other’ with which we compare ourselves—in virtual workgroups or the lack of an appropriate referent in a multicultural or a multilingual group across the globe affects motivation and productivity of group members [20]. A study of a multicultural and multilingual group found that effective virtual group managers perform multiple leadership roles simultaneously [47]. They act as mentors, exhibit empathy, are assertive without overbearing, articulate role relationships, build relationships and trust, and provide detailed and regular communication with the group members. These managers schedule periodic meetings anticipating task complexity instead of waiting for a crisis to warrant one. They also implement plans and procedures in a timely and coherent manner. In addition, they define the interaction protocol for guiding the group behavior in the virtual environment [18], and ensure prompt interactions in asynchronous
communications. When managing a virtual workgroup, certain types of management behavior have positive impact on performance, while other types of management have negative impact on performance [70]. For example, when managers avoid managing conflicts or when they make compromises in managing conflicts, the group performance declines. When the managers create competition and collaborate in managing conflicts, the group performance improves.

Project management in OSD involves increased communication and coordination due to distance, time, and domain expertise differences in onshore and offshore countries [11]. Since successful project management involves successful management of people, the project may need a manager from the offshore facility, but it lengthens the channel of communication and complicates the interactions with the onshore users. Loss of control is a major disadvantage in local IT outsourcing [33] and it will exacerbate in OSD, should the offshore company manage the project. Allocation of resources such as personnel may become bureaucratic, involve time-consuming communication and coordination, and increase the task coupling besides reducing the group cohesion. If the project manager is from onshore, then managing personnel of the offshore facility becomes complex due to language and culture that raise barriers to communication, and reduce group cohesion. Since most project management activities in OSD need F2F interactions, the project manager needs to travel between sites and stay close to the place, where task couplings are high or group cohesion is low.

Researchers have just started studying the management of projects using newer development methods such as ADMs [12]. The lack of effort estimation and project management tools with ADMs complicates the staffing, scheduling, and project control issues in OSD projects. In addition, inadequate documentation in these methods creates additional task complexities in managing the OSD projects. Furthermore, the dependence of ADMs on self-managed teams, and the separation of team members in time and distance increase the task coupling in managing OSD projects.

7. Virtuality depending on system characteristics

The group work in OSD has varied virtuality levels depending on the system characteristics such as the strategic importance of the system, technology, system complexity, system size, system structure, and differentiation of business processes in the system.

Virtuality in development of strategic systems. Technology innovation and diffusion in a company follow four phases [7]. In the first phase, new and untested technologies undergo investigation and experimentation by IT people. In the second phase, users and IT people jointly develop pilot projects using the new technology. A company that develops systems as a competitive necessity, or as a strategic weapon to reduce cost or to differentiate its products and services from that of its competitors is likely to use technologies in these two phases. Innovation, creativity, and experimentation during the first two phases of technology diffusion involve users, new technology that is not widespread, and pilot projects in collaboration with users. Since strategic systems tend to be new applications, users may not have a good picture of their requirements. The development of strategic systems, which involves knowledge-intensive collaborative work with unclear objectives and tasks, has high task coupling.

Most offshore software companies spend very little on research and development [34]. In addition, they may not have the necessary skills in phase one and phase two technologies. Most software companies in developing countries have a skewed profile of skills with more of programming expertise than analysis and design [36]. Developing countries produce a large number of IT graduates, but very little IT teachers. Therefore, these countries may not have skills in the latest technologies and development processes [34]. The global virtual group, separated by time and distance, may lack a common understanding of the project, and its goals and objectives, which in turn, would lead to task conflicts, process conflicts, and relation conflicts. Software companies in developing countries are unlikely to be familiar with the current and evolving technologies and business practices that become the core of strategic information systems. Besides, virtual work in OSD will involve culturally diverse groups. As discussed under Section 3, different backgrounds, experience, skills, role perceptions, application domain knowledge, IT skills, and mental schema of users and analysts reduce the group cohesion. Depending on other group member characteristics, the cohesion for a strategic systems development group would be medium or low.

Unclear requirements and new technologies make the waterfall model unsuitable for offshore developing strategic systems. Although the use of spiral model is uncommon in the development of business information systems, the strategic importance of a system warrants detailed planning, experimentation, verification, validation, and risk management provided by the spiral model. The low cohesion and high task coupling during the prototyping, requirements analysis, and implementation stages have very low virtuality levels that need F2F interactions. Through F2F interactions in the earlier stages, the group cohesion can increase to a higher level at later stages. The use of prototype, shared application development data repository, and group support systems can reduce the task coupling during user interface design to moderate level. Combined with improved group cohesion, the virtuality level of user interface design would be low/medium. Strategic systems are likely to have uncommon and new business processes, whose requirements may evolve even during coding and testing. However, the use of a prototype at each stage could limit these
changes, reduce the task coupling, and increase the virtuality levels to medium/high during other design, coding, and testing stages. Coordination of various activities during systems development is often a major problem that contributes to the project failure [53], and the magnitude of the problem increases for strategic systems developed in dispersed locations. As discussed earlier, the project management of strategic systems calls for user control with a steering committee and has a high task coupling and a very low virtuality level.

Strategic systems can also be developed by the combination of prototyping and waterfall model. The prototyping helps to define clear objectives, requirements, and resolve technical uncertainties associated with the development of strategic systems. The virtuality levels would be similar to those associated with the spiral model with an exception. Since the requirements may evolve even during coding and testing, other design activities, coding and testing stages would have moderate task couplings and low/medium virtuality levels.

The RAD approach is suitable when the strategic systems development has unclear objectives, changing requirements, and new technologies, but the high task coupling and low group cohesion in OSD makes it unsuitable. As discussed earlier, the requirements planning and cutover stages have high task couplings and very low virtuality levels. Since the group cohesion can improve during the iterative design and construction stages, the design stage with its moderate/high task coupling would have a medium/low virtuality level, and the construction stage with its low/medium task coupling would have a high/medium virtuality level. The project management has very low virtuality as with the above approaches.

The use of ADMs in the development of strategic systems in OSD would have very low virtuality level because of high task coupling through all phases of development and low group cohesion exacerbated by the lack of domain knowledge that is essential for developers in ADMs. The lack of methods and tools for effective effort estimation, planning, control, and documentation in iterative and incremental development creates high task coupling. This, combined with low group cohesion in self-managed teams separated by time and distance, creates very low virtuality level.

The suitability of global virtual work in the development of non-strategic systems depends on technology, system complexity, structure, differentiation in business processes underlying the system, and size.

Virtuality in development of complex and high technology systems. When developing a system using high technology, users may lack knowledge of the technology and the application of such technologies to their business functions [71]. In addition, the offshore developers may lack knowledge of the application area, user requirements, user expectations, and the extent of new application integration with current systems. High technology, inexperience with technology, and complexity in technologies create uncertainties in both development and implementation [7]. The development process would have high task coupling because these uncertainties create complexities in the group process, increase the interaction frequency, and message complexity.

Systems with high technology and complexity can be developed using a combination of prototyping and the waterfall model. The prototype helps to solve technical uncertainties and to explore integration issues before a full-fledged development. Combined with low group cohesion and high task coupling, both prototyping and requirements analysis stages have very low virtuality levels. Better group cohesion through F2F interactions in the earlier stages and moderate task coupling with the use of prototype, shared application development data repository, and group support systems, the user interface design would have medium virtuality level. Other design, coding, and testing stages have high virtuality levels. Since coordination of various activities during systems development increases with system complexity [53], project management for high technology and complex systems in OSD environment would also have high task coupling. The management of a project with high technology and complexity needs IT leadership, and has a low virtuality level. The project management should involve scheduled F2F meetings with developers to ensure the resolution of technical issues and the development progress per schedule. The virtuality levels with the use of RAD and ADMs in OSD are as with the strategic systems.

Virtuality in development of low structure systems. The structure of a system relates to the extent of changes that may occur in scope, complexity, technology, processes, or organization during system development [7]. Systems with low structure have many changes during development, and systems with high structure have few changes during development. Lack of top management commitment, failure to gain user commitment and involvement, and conflicts between user departments create user-oriented changes that affect the systems development [49]. Inadequate user involvement is often a major cause for project failures. If the proposed system replaces most of the current functions, calls for major procedural changes, and involves multiple departments, then conflicts between users are likely to be high. In such systems developed in-house using F2F interactions, the focus of project management shifts from managing the tasks to reconciling user differences. Culturally diverse groups in OSD would have more conflicts, and need increased conflict management. Trust among group members can reduce conflicts, but trust is difficult to establish in virtual groups for OSD of such systems. These characteristics of the virtual group constitute low cohesion in the OSD of low structure systems.

Changing objectives, technology, and requirements create scope-oriented changes [49]. Such changes may occur because the users have no agreement, the users are
unsure of their requirements, identified requirements are incorrect or incomplete, or the user requirements are changing with evolving business needs. Scope-oriented changes can compromise quality, and escalate cost and time due to increased coordination, communication, problem resolution, user management, and change management. In OSD, the management of scope-oriented changes involves high task coupling. The management of scope-oriented changes needs appropriate requirements gathering methods and user involvement. Verification of requirements with user involvement is essential to define correct and complete requirements. Developing a prototype and validating it with users before full-fledged development can reduce the scope-oriented changes.

Combined with low group cohesion, systems with low structure have a very low virtuality level during prototyping and requirements analysis. Although the prototyping can help manage scope-related changes for non-strategic systems, the user-oriented changes will create low virtuality level for the user interface design. Other design activities, and coding and testing stages would have moderate task couplings because objectives, technology and requirements may continue to change. If requirements can be frozen, the task coupling would be low. Combined with low group cohesion that is a characteristic of the development of low structure systems, these activities would have low/medium virtuality levels. In low structure systems, project management activities have high task coupling, low group cohesion, and a very low virtuality level. In managing low structure projects, the project manager should have the authority to decide the time schedule, commit users and developers to the time schedule, and override scope creep after user sign-off of the requirements definition. The project manager should also have sufficient authority and influence to ensure user participation, and to resolve differences between user departments. Please note that low structure systems combined with high technology and complexity have higher task couplings, lower group cohesion, and lower virtuality levels.

The RAD and ADMs are suitable for developing low structure systems in an F2F environment, but they have the same limitations of strategic systems in OSD.

**Virtuality in development of systems with differentiated business processes.** Although differentiated business processes in non-strategic systems add to costs with no corresponding benefits, they are commonplace in many companies. Offshore companies may not have knowledge or expertise with such business processes. The requirements analysis stage and the verification of requirements in the OSD environment for such a system with uncommon and new requirements would have a high task coupling. Combined with medium or low group cohesion, the requirements analysis stage has a very low virtuality level. The use of a prototype, shared application development data repository, and group support systems can reduce the task coupling during user interface design to a moderate level.

Combined with improved group cohesion, the user interface design would have low/medium virtuality level. Since differentiated business processes have no benefits in non-strategic systems, the freezing of requirements after user interface design would create low task coupling and medium to high virtuality levels for other design activities, coding and testing stages. The project management has a high task coupling and creates a low virtuality level. If requirements can be frozen, the waterfall model that provides modularity is preferable to the incremental and iterative approaches that have high task couplings in all phases of development.

**Virtual workgroups in development of large systems.** A system is considered large if the offshore company has no experience in developing systems of such sizes. For example, to a company that has never developed a system with greater than 100,000 lines of code, a project with 200,000 lines of code is large. Excepting the requirements analysis that has a low virtuality level and user interface design that can have a medium virtuality level, other stages of high structure and non-strategic systems with common business processes and technologies have high virtuality. The incremental and iterative approaches that have high task couplings in all phases of development have lower virtuality levels than the structured approach for such systems. Since coordination of systems increases with size, and the company has had no experience with large projects, the project management would have high task coupling. For a group with high group cohesion, the project management would create a low virtuality level. Table 3 summarizes the virtuality levels for various types of information systems and development approaches.

**Virtuality depending on the place and type of offshore facility**

The choice of a country for the offshore facility depends on the availability of right skills at low cost and a good communication infrastructure. Choosing certain offshore countries limits the use of synchronous communication media because extended time zone differences between onshore and offshore countries require either party to work at odd hours. Extended time zone differences are acceptable for development work that can be done asynchronously, but that too places a burden on the project manager. Offshore facilities in countries that provide for adequate protection of intellectual property rights may have difficulties in earning the trust that affects group cohesion and virtual work. Previous sections discussed the effectiveness of virtual work in relation to diverse cultures and languages.

An onshore company can contract with multiple foreign vendors to have alternate sources for fallback, but it requires expensive synchronous and asynchronous communication facilities with each offshore facility. Unless the onshore company places large and frequent contracts, the vendor may not be motivated to be responsive to the company = s
needs, which can reduce the group cohesion and increase the task coupling. In projects with incomplete contracts, the objectives of the onshore company that tries to maximize its savings conflict with the objectives of the offshore vendor that tries to maximize its profits. Incomplete contracting includes inadequate and incorrect terms of contract, ambiguous definitions in service level agreement, inadequate definitions of system functions, deliverables, quality, and schedule. Trust in OSD is more important than other virtual works because having a well-written contract to cover all uncertainties in systems development projects may not be possible [51,91]. The contract management could evolve into a major issue in such OSD projects to increase task coupling and reduce group cohesion. The contract management has a low virtuality level that should be managed by frequent F2F interactions and rich synchronous communications.

Startup companies and other companies that do not have the maturity and experience in IT outsourcing may lack the communication infrastructure, virtual interactions, documentation, development processes, and management systems, and promote group cohesion.

8. Future research and conclusion

This article discussed that the effectiveness of virtual work in OSD varies depending upon the system development processes, project management methods, a system’s strategic nature, various system characteristics, and the type of offshore facility. It discussed the development activities and management procedures amenable to virtual work in OSD projects, and a framework to use virtual work selectively while offshore developing various types of information systems.

Although organizations claim various advantages, cost considerations primarily motivate OSD. The costs of OSD include the production costs, and the transaction costs related to sourcing, contracting, and the management of source, contract, project, and risks. Organizations consider production costs six times more than transactions costs in local outsourcing projects [6], but the transaction attributes affect the outsourcing success [106]. Organizations that consider just the production costs and ignore various transaction costs in OSD may not achieve their cost saving objectives [10,31]. Since a study of just two actors separated by time showed that coordination costs are higher than when they work about the same time [28], we can expect higher transaction costs with larger groups separated by both time
and distance. Ineffective virtual workgroups increase the transaction costs and development time, and compromise the quality.

Available development effort estimation methods assume work in colocated places. Methods to estimate development efforts in the OSD environment are necessary to determine the cost savings in offshoring. In addition, research is needed to evaluate management methods in OSD projects that help achieve cost savings without sacrificing time and quality.

Virtual work is necessary for the success of OSD, but it has several limitations in the OSD environment. Since rich synchronous and less-rich synchronous communication media are used as a substitute for F2F interactions, empirical research is needed to determine the effectiveness of using these media vis-à-vis F2F for various systems development activities. New methods of development in the OSD environment can be evolved based on the effectiveness of various interaction media, virtual work protocols, and group support systems.

Structured approaches have been subjected to several decades of empirical research on various issues, but newer incremental and iterative development approaches lack empirical evidence to support various claims of their proponents. The effectiveness of iterative and incremental approaches for various types of systems needs to be examined. These approaches also need methods to estimate development effort and to manage projects.

Organizations have been trying to use current development processes—many of which have varied success in the F2F environment—in a virtual environment. Instead of retrofitting these development processes to a virtual environment, new systems development processes are needed for OSD. Until such time, OSD managers should understand the limitations of virtual work, and use virtual work selectively to benefit from OSD.

References


