human relations
THE TAVISTOCK INSTITUTE

human relations
63(8) 1199–1221

DOI: 10.1177/0018726709355657
hum.sagepub.com
SAGE

Boundary object use in cross-cultural software development teams

Michael Barrett
University of Cambridge, UK

Eivor Oborn
Royal Holloway University of London, UK

Abstract
This article examines the evolving use of boundary objects in cross-cultural software teams. Our field study of a Jamaican-Indian team examines the use of software specifications and project management tools as boundary objects in facilitating sharing across knowledge boundaries. We examine how and why the role and use of boundary objects may facilitate collaboration across knowledge boundaries at one time and contribute to conflict at other times. We unpack the interacting elements that both facilitate and constrain knowledge sharing, and trigger conflicts at different stages of the software team development. Specifically, we found that the use of boundary objects at transitions involving definitional control and the subsequent redistribution of power/authority may inhibit knowledge sharing. The subsequent reifying of cultural boundaries along with negative stereotyping led to relational conflict, through a process we call *culturizing*, as cross-cultural differences emerged as problematic for team dynamics.

Keywords
boundary objects, collaboration, conflict, cross-cultural, information systems development, knowledge, software development, teams

Introduction

Over the last couple of decades, the concept of boundary object (Star and Griesemer, 1989) has been utilized across a range of fields from new product development (Carlile,

Corresponding author:
Mike Barrett, Judge Business School, University of Cambridge, Trumpington Street, Cambridge CB2 1AG, UK.
Email: m.barrett@jbs.cam.ac.uk

2002), innovation (Swan et al., 2007), and project management (Sapsed and Salter, 2004; Yakura, 2002). Boundary objects are mediating artefacts that have interpretive flexibility and can be an important means of achieving collaboration, promoting the sharing of knowledge between diverse groups (Sapsed and Salter, 2004).

The practice of software development incorporates numerous objects including requirements specifications (specs) and project management methodologies. These objects seek to bridge or overcome knowledge boundaries (Carlile, 2004) between those with technical knowledge and others with domain specific knowledge (Tiwana and Mclean, 2005). In the process, knowledge sharing is critical in the coordination of expertise among team members (Faraj and Sproull, 2000). However, structural diversity (Cummings, 2004), embeddedness of knowledge, and status differences (Metiu, 2006) between team members can raise obstacles to effective collaboration (Levina and Vaast, 2008).

While the boundary object (BO) literature has primarily focused on the positive and collaborative role of objects in bridging knowledge boundaries (Ewenstein and Whyte, forthcoming; Swan et al., 2007), rigid shared objects do not facilitate effective integration of tacit knowing in diverse teams (Levina and Vaast, 2006; Vaast and Levina, 2006), and may limit learning across groups. Carlile (2002) highlighted this multi-facetted role of objects and points out that boundary objects are not magic bullets as their characteristics are hard to sustain as problems and people change; as we highlight this may also be the case when they are used by the same team members over time.

Our conceptualization of boundary objects gives less focus to their essential characteristics as 'monolithic autonomous edifices' (Harvey and Chrisman, 1998) and more to the negotiation processes that go into transforming an artefact into a boundary object. Complex processes of maintaining the boundary object (Thomas et al., 2008) occur through the localized involvement of artifacts and people. Thomas et al. (2008) highlighted that boundary objects are permeated with power. This is consistent with Star and Griesemer's (1989) original conception of boundary objects where they noted the potential for its imposition given actors' unequal access to resources and authority (Briers and Chua, 2001) and the possible consequences for silencing and fragmentation (Levina, 2005).

We contribute to the growing literature on boundary objects in two ways. First, we build on an interactional perspective of boundary objects to better understand how and why their role and use may facilitate collaboration at one time and contribute to conflict at other times within cross-cultural software teams. We pay close attention to the political dynamics associated with object use, as has recently been advanced in the literature. Second, we connect the boundary object literature with cross-cultural research in the context of software development teams. While literatures on boundary objects and software development have focused on diversity of workgroups across knowledge boundaries, we additionally examine cultural diversity within these teams (Barrett et al., 1996; Walsham, 2002). Advances in technology, globalization, and labour mobility have significantly increased the likelihood of employees working with others of different cultural backgrounds (Randal, 2003), which can pose a challenge for software teams in completing projects on time and on budget (Faraj and Sproull, 2000) given the potential of nationalism for relational conflict (Ayub and Jehn, 2006).

Literature review

As Table 1 illustrates, we start by highlighting the functional and symbolic elements of objects in information systems development (ISD), which may be used in certain contexts as boundary objects. We subsequently discuss boundary objects' different types and functions. As shown in the third row of our table we close this section by highlighting the potential for conflict within cross-cultural ISD teams.

Objects in information systems development (ISD)

ISD is a complex activity that involves people of diverse skills who engage with numerous objects and artefacts (Levina, 2005; Levina and Vaast, 2005; Wastell, 1999). A key challenge is to specify and coordinate the work amid numerous stakeholders (DeSanctis and Jackson, 1994; Faraj and Sproull, 2000) having diverse knowledge domains and coordinating across social boundaries (Kraut and Streeter, 1995).

Requirement specifications are at the centre of the software development process as diverse groups use this to stipulate the functionality required of the system in developing code. The nature of requirement specifications are often incomplete owing to the uncertainty related to changes in functionality over time as business needs and users' desires evolve (Kraut and Streeter, 1995). Furthermore, incompleteness can result from the limited nature of domain knowledge that is critical during design. The lengthy indeterminate process of developing requirement specifications is often 'shut down' before a full understanding can be achieved (Walz et al., 1993).

Not only do specifications help to articulate a shared understanding of the application domain between developers and users, but as transitional objects they facilitate learning by instilling confidence and a sense of control (Wastell, 1999). It has been argued that the lack of ability to learn is a source of failure for many ISD projects (Lyytinen and Robey, 1999; Wastell, 1999), which may stem from the inadequate use of objects. In addition to the specification (spec), other objects used in ISD practice are timelines, Gantt charts, and project management techniques (Henderson, 1991; Yakura, 2002). These objects are not just instrumental tools used to visually manage resources, but also promise closure to an inherently uncertain endeavour. This symbolic role of objects highlights their multidimensional nature (Ewenstein and Whyte, 2009).

Another role that objects can play is to highlight status inequalities within the software development process (Levina and Vaast, 2008; Metiu, 2006). Perceived status differences between members influences their willingness to cooperate. Closure between groups, for example through geographical or ethnic boundaries, can heighten alienation and stymie collaboration during coding. ISD tools and methodologies are not only repositories of knowledge but embody authority as to how things should be done and can act as effective boundary objects (Henderson, 1991).

Boundary objects

Boundary objects have been defined as being 'both plastic enough to adapt to local needs' and 'robust enough to maintain a common identity across sites' (Star and Griesemer,

Table 1 Objects and knowledge boundaries in cross-cultural information systems development teams

	Key concepts	Key references
Objects in ISD	*Functional elements*	
	- Coordination of knowledge and expertise	Desanctis and Jackson (1994), Faraj and Sproull (2000)
	- Requirements spec and project management tools as 'transition' objects for learning in ISD	Wastell (1999)
	- Multidimensional nature	Ewenstein and Whyte (2009)
	Symbolic elements	
	- Highlight status inequalities	Levina and Vaast (2008)
	- Authority over objects can reinforce or redistribute task area boundaries	Sapsed and Salter (2004)
Boundary objects	*Types*	
	- Diverse types facilitate sharing across groups	Star (1989), Star and Griesmer (1989)
	- Concrete, visionary, ideal	Carlile (2002), Briers and Chua (2001)
	- Include ISD tools such as project management, specs	Henderson (1991)
	- Designated or in-use	Levina and Vaast (2005)
	Function	
	- Timelines are interpretively flexible, and as temporal BO offer closure as illusion of control	Yakura (2002)
	- Capacity to adapt to local dialects and be flexible to localization	D'Adderio (2004), Lindkvist et al. (1998)
	- Not easily substituted with other objects	Henderson (1991)
	- Suitability for different knowledge boundaries including pragmatic boundaries which can arise from vested interests	Carlile (2002, 2004), Swan et al. (2007)
	- Represent jurisdictions and status	Bechky (2003)
	- Reinforce boundaries and authority	Levina and Vaast (2006)
Cross-cultural working and ISD teams	- Conflict in ISD owing to lack of user involvement; engagement; power and status dynamics	Levina and Vaast (2008), Markus (1983), Metiu (2006), Robey (1989)
	- Cultural diversity contributes to task and relational conflict in ISD	Barrett et al. (1996), Walsham (2002)
	- Knowledge sharing challenges in cross-cultural working	Ford and Chan (2003), Peltokorpi (2006)
	- Multiple overlapping boundaries give rise to status differences and inhibit collaboration	Levina and Vaast (2008)
	- Cultural boundaries as product of social processes	Wimmer (2008)

1989: 393). They facilitate the sharing of knowledge between diverse groups (Sapsed and Salter, 2004) by both facilitating an overlap of meaning yet preserving sufficient ambiguity as groups read their own meanings (Star, 1989). A number of different boundary objects have been studied. Briers and Chua (2001) distinguish between visionary boundary objects, which are conceptual in nature (e.g. organizational 'best practices'), and ideal boundary objects, while Carlile (2002) emphasizes the importance of concrete boundary objects. All objects are not automatically boundary objects even if they are designated as such (Levina and Vaast, 2005), but are brought to life through social interaction (Briers and Chua, 2001) and in use.

In addition to boundary object types, the literature has distinguished their function in different contexts. For example, Yakura (2002) suggests that timelines as temporal boundary objects are interpretively flexible by different groups, and offer closure as an illusion of control over a project. Her analysis of timelines during IS implementation gives a compelling account of how graphical representations of temporal units operate as a boundary object that remains abstract in use yet able to reconcile diverse temporal arrangements. Others have highlighted the capacity of timelines to adapt to local dialects and be flexible to localization (D'Adderio, 2004; Lindkvist et al., 1998).

Concerning the ability of different boundary objects to function effectively in groups, Henderson's study (1991) underscores how substituting one boundary object (sketches) for another (computer aided design) can hamper the flexibility needed between groups to share knowledge effectively. In addition to what the use of boundary objects does in different settings, Carlile (2002, 2004) has highlighted that their properties develop in respect to different types of knowledge boundaries that exist within a project team. The most basic type of knowledge boundary is syntactic, where informational dissimilarities exist. Semantic boundaries are constituted by the interpretive differences between social groups where meaning is developed from within a unique cultural perspective. Pragmatic boundaries arise from the vested interests held by diverse groups. Boundary objects are not able to span these diverse boundaries equally (Swan et al., 2007). For example, a database may be most useful in spanning syntactic knowledge boundaries, while information packs that require the engagement of numerous stakeholder groups to develop around their vested interests are useful in bridging pragmatic boundaries (Swan et al., 2007). Boundary objects can also mediate relations between different groups and protect work jurisdictions (Bechky, 2003) while reinforcing boundaries and authority (Levina and Waast, 2005).

Cross-cultural ISD teams

The software development literature highlights the ongoing difficulties in completing on time and to budget (Faraj and Sproull, 2000) and points to the frequent conflict that arises during the development process (Robey et al., 1989; Wastell, 1999). Conflict can occur within the ISD team (intragroup), or within the organization (intergroup), for example, when users resist implementation (Barki and Hartwick, 2001). Conflict may be owing to inappropriate user involvement (Barki and Hardwick, 1994), lack of group ownership (Metiu, 2006), or poor management practices (Cohen et al., 2004). The complex and

stressful environment surrounding ISD projects amplify social defences as described in the previous section (Wastell, 1999), opening the door to conflict.

Conflict in groups is commonly distinguished as task, process, or relational (Jehn and Mannix, 2001), though these frequently overlap in reality (Korsgaard et al., 2008). The role of cross-cultural issues in software development is being increasingly recognized (Barrett et al., 1996; Walsham, 2002) with increased labour mobility and the proliferation of global work arrangements (Levina and Vaast, 2008). Organizational and national culture can significantly influence how people view expertise and information (Leidner and Kayworth, 2006). Status issues associated with software coding has been noted to influence social boundaries (Levina and Vaast, 2008; Metiu, 2006). A key difficulty in cross-cultural working is sharing knowledge between social groups, though this issue has received little attention to date (Ford and Chan, 2003; Peltokorpi, 2006) and knowledge exchange is a key aspect of software development (Faraj and Sproull, 2000).

The boundaries between culturally diverse groups gain salience in specific contexts (Wimmer, 2008), and these cultural boundaries are the product of social processes, which are made and remade depending on the circumstances (Wimmer, 2008). We suggest that research on conflict in culturally diverse IS teams not only go beyond understanding conflict in terms of cultural incompatibility (Ford and Chan, 2003; Holden, 2001), but also better understand how and why boundary objects use have implications for knowledge sharing across knowledge boundaries in cross-cultural software development.

Theoretical approach

Conceptualizing boundary objects

Our interactional perspective foregrounds how boundary objects are brought to life through social interaction as diverse actors negotiate collective meaning through and around these objects. Rather than ascribing the boundary object with essentialist properties, we adopt a relational view that highlights that an artefact only becomes a boundary object in use (Levina and Vaast, 2005). Objects and the interactions they mediate are always in a state of becoming (Tsoukas and Chia, 2002), and promote sharing of knowledge across different groups (Sapsed and Salter, 2004).

As initially developed by Star (1989), boundary objects enable collaboration because they are able to facilitate the spanning of knowledge boundaries between diverse occupational communities. These objects are riven with tension and ambiguity (Thomas et al., 2008), allowing enough overlap of meaning to make the object recognizable to different groups, yet ambiguous enough to allow flexible interpretation within contexts inevitably challenged by diverse social arrangements. In this dynamic view, we highlight that negotiations and knowledge sharing enabled by a boundary object at one point in time can change, though the object itself may remain unchanged. Boundary objects can also interact with other objects and influence how these are perceived and used. Ongoing negotiations around one object can control or mitigate the use of related objects.

Furthermore, we adopt a pluralist view of boundary objects that goes beyond a unitary focus on their collaborative potential and calls attention to the political dynamics and conflict that can ensue in their production and use (see Thomas et al., 2008). How objects are operationalized is constantly being (re)negotiated and contested during interactions and across organizational processes. Stakeholders and social groups have differing access to resources, authority and definitional control of the object, whereby dominant actors can impose a particular use of the objects thereby influencing relational and task processes (Levina and Vaast, 2008).

Tacit knowing and boundary object use across knowledge boundaries

Our theoretical perspective also draws on Polanyi's view of tacit knowing (1968), which is consistent with and complements our interactional approach to boundary objects. His focus on tacit knowing is helpful in examining how different groups draw on numerous clues in an interactional manner to integrate and derive meaning. Polanyi suggests that tacit knowledge is 'the outcome of an individual's active shaping of experience' (p. 6) and draws from individual values and dispositions. People derive meaning and communicate their understanding through tacit integration of multiple conflicting clues. Clues can be diverse, consisting of events, written information, or visual details. In ISD, analysts tacitly integrate an understanding of client-user needs for an information system and then explicitly articulate this knowledge through writing a spec. These processes repeat themselves when programmers then code a system from the spec. Those programmers who draw tacitly from different work practices and industry experiences will arrive at different interpretations of the spec's meaning. Thus a part of a programmer's knowledge is not just in the spec, but also in understanding how to use the spec in practice and involving users in system development. These components of tacit knowing are central to making the explicit knowledge encoded in the spec actionable as tacit knowing lends meaning to explicit knowing and controls its uses (Polanyi, 1968).

Tacit knowing provides a basis for the interpretive flexibility of boundary objects and their ability to be plastic in use. Different aspects of an object can be retained as a focus while other aspects may become subsidiary. Polanyi shows how tacitly integrated meaning shifts when clues move from focal to subsidiary positions, such as when a pianist shifts focus from the melody to a focus of moving their fingers (the latter of which can make the player draw a blank). Learning to focus on, or keep subsidiary, aspects of objects can be developed within social contexts; graphic designers may focus on certain features of a webpage while technical specialists may keep other features focal, rendering the object to be used in different ways (Levina, 2005).

In sum, our theoretical perspective adopts a pluralist and interactional approach to illuminate the role and use of boundary objects across diverse knowledge boundaries. This conceptualization of boundary objects recognizes their collaborative and political capacity, as their use is constantly being negotiated and contested between artifacts and other people through interactions over time. Further, Polanyi's view of tacit knowing deepens our understanding of the process whereby members draw on functional and

cultural clues of both objects and people, once again in an interactional manner to derive meaning across knowledge boundaries.

Research methodology

Data collection and analysis

Case studies are the preferred research strategy for a process study when an in-depth understanding of phenomena is needed (Eisenhardt, 1989). We conducted a two-year longitudinal case study of cross-cultural software development for a general insurance system across three phases. We began the study to explore the role and use of IT for business innovation in the Jamaican insurance industry.

In the first phase of the study, we focused on the macro-level context of the global reinsurance crisis. Local insurance firms in countries such as Jamaica have low levels of insurance capital or reserves yet are exposed to perils of hurricanes and earthquakes, and have tended to rely on reinsurance. This involves the transfer of some or all of an insurance risk from the local insurance company to another typically large global insurance company. We studied the sectoral initiatives triggered by the reinsurance crisis around geographical information systems and hazard mapping and the development of a new insurance system, GENSYS, for the general insurance firm, GENSURE, within the financial conglomerate, JAMSURE.[1] We drew on a number of secondary sources collected from sectoral studies, trade journals, and local newspapers. In addition, 12 interviews were held with the Insurance College of Jamaica, the regional housing and urban development office, and insurance companies leading the sectoral efforts.

Data collection at the organizational level started at the end of the first phase when access was negotiated through the chairman of JAMSURE. Our analysis of the initial set of interviews highlighted an intriguing and unique aspect of the case concerning the use of a cross-cultural software team to meet the unprecedented needs of global reinsurers. In total, 42 interviews were conducted over three phases with a wide range of participants: senior managers of JAMSURE and its sister companies in IT, life, and general insurance company; project managers, team leaders, and developers at the IT and general insurance companies; and intended users and managers of the GENSYS system. We conducted repeat interviews with Indian project manager and the Jamaican manager who headed the development effort, and four of the five team leaders of Indian and Jamaican descent. We also interviewed twice both the CEO of GENSURE and of GROUPIT, as well as the Director of Operational and implementation manager at GENSURE responsible for implementing GENSYS. The purpose of the repeat interviews was to examine changes associated with the ongoing implementation and redevelopment of GENSYS and to explore underlying reasons for the cross-cultural issues that emerged over time.

Detailed notes were taken of interviews rather than recording owing to the cross-cultural sensitivity that persisted between Indians at the group's IT company, GROUPIT, and Jamaican staff over the duration of the study. Primary data sources supporting this interview data include textual data such as the conglomerate's group strategy, IS strategy, mission statements, newsletters, and annual reports.

Open coding identified key issues that challenged the formation and ongoing development of the software team. Among others, key themes included cross-cultural attitudes and differences, broader power relations within the team, and reduced levels of knowledge sharing between groups over time. A further step in the coding analysis was the identification of recurring themes from across these broad categories over the time period studied.

In a final step, we drew on different theoretical perspectives to think about the information (Langley, 1999). Specifically, we drew initially on Giddens's theory of globalization and the coordination in ISD literature to enrich our performative account of the change process and the macro-micro interdependencies, which were critical to the research setting. We also drew on theories of knowledge to deepen our understanding of the micro-processes of knowledge sharing in this cross-cultural context.

Case description

Prior to the reinsurance crisis of the 1990s, locally owned firms experienced rapid growth and good profitability following deregulation of the general insurance industry. During the reinsurance crisis, global reinsurers demanded new levels of detailed risk information across different geographical areas to facilitate variable pricing of risk.

This placed significant pressures on the Jamaican insurance sector to develop novel geographical information systems to support the redesign of local firms' insurance systems. JAMSURE, the largest local insurance conglomerate, viewed this challenge positively as an opportunity for growth in meeting its aspirations to be a Fortune 500 company. JAMSURE was a highly diversified financial group of 27 companies that had grown through acquisitions and had revenues of approx USD 400 million. Along with his chief financial officer and group HR manager, the JAMSURE chairman oversaw the genesis of the new general insurance system (GENSYS), being developed to meet reinsurers' changing information requirements in a timely manner.

After initial attempts to acquire a system that met their requirements failed, the decision was made to develop a state-of-the-art system in-house. The Jamaican office of a Big Four consultancy firm, GLOBAL, was commissioned to work closely with JAMSURE's top management and a user group in developing the functional specifications and initial design of GENSYS. During this interaction with GLOBAL, which had successfully hired the vast majority of its consultants from India, JAMSURE took the decision to follow suit and establish their own software company, GROUPIT. The for-profit software subsidiary would be responsible for bidding for the later phases of GENSYS development and provide services both internally across the JAMSURE group and externally across Jamaica and regionally. Dr Prava and a number of experienced developers were hired from software houses in India to form the top management and team leaders of GROUPIT. Dr Prava, a former president of an Indian software house, had a strong technical background and a PhD in Operations Research from a top US university.

GROUPIT bid for and successfully won the tender for the software development of GENSYS based on GLOBAL's spec. Following attempts to code from the spec, short-term consultancy help was reluctantly solicited from GLOBAL to 'add to' and 'fill in' the spec as 'details were residing in someone's head there'. However, further attempts at

system development reinforced the need for additional local general insurance expertise in order to write the code and build the system. A team from GENSURE's management information system (MIS) department was seconded to join GROUPIT. These included the Jamaican MIS manager along with carefully selected staff who were tested and demonstrated considerable aptitude for software development. The cross-cultural software team comprised the project manager who was the Indian CEO and technical director of GROUPIT, the Jamaican MIS manager from GENSURE, and five teams of five developers. Three of the teams had a Jamaican team leader and two had an Indian team leader, each managing a team of two Indian and two Jamaican developers. However, they were now two months behind schedule.

Despite GENSURE's late involvement in the development process and their initial feelings of being excluded, genuine excitement and enthusiasm developed across this culturally diverse team. Learning professional software development skills using state-of-the-art system technology was very enticing for the Jamaican group, as were GROUPIT's ambitions of marketing the software internationally. A culture of enthusiasm and knowledge sharing abounded across the software team with weekly awards for the most helpful member and project champion. There were positive reputation effects that gave hope and encouragement to the project participants. In particular, a New Zealand insurance company short-listed GENSYS in an international search for a flexible reinsurance system and a large multinational IT vendor entered into agreements to market the system globally.

However, approximately half way through its nine-month development cycle, problems surfaced leading to conflict as deadlines tightened, with Jamaicans complaining of a highly competitive blame culture and resentment of rigid weekly project deadlines. Dr Prava was perceived to be 'precise and very scientific', and relied heavily on his most talented technical Indian project manager both for his technical knowledge and deployment of project management methods. Such a focus on formal coordination practices and perceived lack of participation by Jamaicans plagued the development effort. User involvement became minimal and any feedback outside of the spec was noted but not incorporated in an attempt to meet the tight project deadlines and provide the 'contracted' deliverables. At the end of the research period, GENSYS had not been successfully implemented despite significant redevelopment efforts for two years after the initial date of delivery and GROUPIT as a team was disbanded.

Case analysis

We present our findings across different phases of the IS development process. Our case analysis examined the interactions between boundary objects and their use by different actors, and recognized the dual potential for collaboration and conflict. In the early phases of IS development, we found that the requirements specification and flexible use of timelines facilitated knowledge sharing in a collaborative manner across the team. In the subsequent phase, the project manager asserted managerial authority and control in his operationalization of timelines, which had significant implications for limiting interactions between technical specialists and domain specialists, in their use of the spec in programming. The final phase highlights how task conflict spurred on by

different interpretations of boundary object use adversely affected levels of knowledge sharing and disrupted the team's shared identity. This led to negative stereotyping of different national cultures and subsequent relational conflict between Indian and Jamaican members.

Exploring boundary object use in cross-cultural software development

Phase 1: From initiation to collaboration Following repeated unsuccessful attempts by Indian programmers to develop GENSYS from the spec, the decision was taken to form a cross-cultural team with Jamaican developers from GENSURE. Their initial feelings of participation in working with GROUPIT were mixed; on the one hand, they were given little option, being told by the top management that they 'would have to either sink or swim'; on the other hand, the opportunity for learning and being part of a state-of-the-art software development team was exciting, if a little daunting.

The early use of the spec between Jamaicans and Indians was collaborative with timelines and project meetings being symbolically used to positively reinforce and reward collaboration. Morale was high and there was a shared identity:

> The project started off well. GENSURE and GROUPIT personnel lost identity and all became part of the GENSYS team . . . there were awards for most helpful member and project champion. (GROUPIT developer)

The Jamaican MIS group was excited and valued this opportunity to learn from the Indian developers who were taking the time to share their technical knowledge and skills. In an effort to effectively coordinate the team's expertise, positive incentives were successfully implemented to encourage knowledge sharing, such as weekly prizes to the most helpful member:

> There were other motivations such as a grand prize to Switzerland for the most efficient team as well as incentives (bonuses) for those who met their deadlines. (GENSURE Team Leader)

At this point coordinating expertise did not only rely on sharing task specific knowledge, but also in negotiating how GENSYS would be developed from the spec. All members were able to influence and participate in the process. As one Jamaican explained, 'If there was a problem to be solved, we would sit down and solve it . . . it was a team effort, meet and discuss each project.' As such, the dialogue would centre on the spec, but was not solely determined by the spec, which was seen to be flexible and partial in directing the programming effort.

The spec was used to support knowledge sharing between Jamaican and Indian developers who initially interacted very closely together. On one hand, Jamaicans could tacitly understand system requirements, given their considerable experience in the local industry and working with users to provide enhancements for the existing insurance systems. However, they did not have state-of-the-art technical skills and knowledge of the programming language in which to code effectively. The successful development of the system depended on GROUPIT's ability to coordinate the

unique technical skills and knowledge of Indian programmers and the expertise of the Jamaican MIS insurance staff on insurance domain knowledge into a meaningful integration. On their own, Indian programmers had been unable to write effective code. A GROUPIT developer summarized the importance of the Jamaican developers' expertise:

> There were delays due to inadequacies of design . . . GENSURE MIS who had a better understanding of what functionality was required along with getting input from a few users played a critical role in quality control and carrying out major reworks.

The Indian programmers and Jamaican IT staff drew on the spec in different ways and pieced together what seemed like 'holes'. Their understanding of local needs was made possible owing to the changing and malleable nature of the spec at this early point in the development process. This required 'plasticity' of the spec as a key boundary object that mediated the knowledge sharing process needed to coordinate expertise around the IS development process. Team members were able to negotiate collective meaning, which allowed knowledge sharing using the spec and project management tools (e.g. timelines) as boundary objects.

Though the team noted cultural differences among members, this did not impede successful collaboration in the early phase. Members developed an understanding of the spec requirements and then drew on each other's knowledgeability to share an understanding that yielded good progress in the software development effort. The spec acted as a facilitating and coordinating device for information exchanges across the cultural and occupational boundaries between workers, even though each group drew on the spec in different ways as undergirded by their expert tacit knowledge.

As a boundary object between the JAMSURE management and the development team, the spec represented a malleable set of requirements that the reinsurers, through the management, used to clarify ways of reducing risk, and the development team used the spec to build a system that would meet local needs. Given this malleability and flexibility, the software team was able to rework the spec to meet the new risk requirements, though over time the pressing deadlines increasingly put strain on the spec as a flexible boundary object in use. In hindsight, several team members referred to this period as their 'honeymoon phase':

> At the start, everything was in a tailspin with a steep learning curve, and everybody was helping each other debug one another's programs . . . after the honeymoon period when morale was high and [there was] excitement about using cutting edge technology a culture clash set in. (GENSURE Team Leader)

Phase 2: Imposed BO use at mid-point transition While the honeymoon phase of the cross-cultural team had finally advanced software development, overall, GENSYS was behind schedule and GROUPIT was already halfway through the nine month contract. The Indian leader imposed tighter deadlines as formal coordination methods in the hope to meet scheduled project deliverables. Jamaican MIS staff expressed difficulty in identifying with and embracing these stringent practices. The prizes and incentives for team

collaboration stopped and Jamaican developers complained that the newly emphasized weekly deadlines were unrealistic. A MIS team leader explained:

> Though teams were (initially) compliant, deadlines [became] rather stringent, if not unreasonable . . . The whole project was conditioned by the strict deadlines imposed weekly.

In asserting his managerial authority and control, the project manager imposed rigidity through a deliberate focus on timelines and Gantt charts, which were mounted in the many offices and meeting rooms. These were symbolically referred to on a frequent basis by the project manager, praised by top management for the confidence and control they provided, and held cynically by Jamaican developers as providing a false sense of security. At the same time, the use of these project management tools as temporal boundary objects decreased attention on the collaboration needed in expertise coordination around the spec and led to an adverse impact limiting team interactions among themselves and with users:

> This is how testing has to be done as they contracted us to produce software based on the specification. Afterwards, if the specifications are changed (after further user input), then we can redo the software. (Dr Prava)

At weekly meetings, Dr Prava increasingly emphasized the need to code strictly from the spec, and would often physically point to the written specification and terms of reference. He emphasized a 'contractual' approach based on building to the pre-defined spec, and could ill afford the time and cost involved with user acceptance, which became subsidiary. This focus on 'coding to the spec' in its current form was perceived to be necessary modus operandi by the project manager as he believed flexibility would likely mean further delays. When the Jamaicans suggested that certain user needs required further clarification, the leader would dismiss their interpretation stating that there was insufficient time to be adding to the spec.

Jamaican MIS staff initially hoped they would learn good programming and project management skills through close collaboration with the Indian programmers. However, missed project milestones put enormous time pressures on both the Jamaicans and Indian developers, squeezing out opportunities for peer level knowledge sharing and perceptions developed around a new level of competitiveness and distrust:

> A lack of trust developed between development teams. It became so competitive . . . competitive behavior demoralized other teams who had invariably not completed their tasks, and felt their competence was indirectly being questioned at progress meetings held weekly. (Jamaican team leader)

In feeling constrained by the project timeline, the leader succeeded in rendering the spec, as a key boundary object, less flexible in use. It no longer had the capacity to act as a facilitating and malleable artifact that could be used in various ways to direct coders to share knowledge and develop meaning. Instead of accommodating work styles and providing an occupational focus in system development, the spec along with mandated tough deadlines now required a more standardized coding practice, which served to limit knowledge sharing:

> The approach taken to writing programs was similar to baking buns (sweet bread) on a conveyor belt . . . the questioning was direct 'have you finished all the programs you are committed to, yes or no' . . . there was not much consideration that we were unfamiliar with state of the art programming and there was a necessary learning curve. (Jamaican IT Staff)

The analogy of a conveyor belt for the development practice suggested an individual and serial effort rather than the earlier interactive practice with individuals building on each other's knowledge and understanding of the spec. The Jamaican staff historically enjoyed close interaction with users in building and designing systems, which contrasted with the traditional compartmentalized way Indian software houses used the spec (Walsham and Sahay, 1999). A Jamaican manager summarized the MIS staff frustration at the privileging of technical knowledge:

> If you talk to users you learn and that is the most important thing to get the person who understands the business functional requirements and not just someone with a million letters after their name (referring to Dr Prava). (General Manager Operations, GENSURE)

Phase 3: BO use and cross-cultural conflict in post-transitional team interaction Relations between Jamaican and Indian developers worsened during Phase 3, as Dr Prava was perceived to rely heavily on the most experienced Indian technical team leader. This served to accentuate the privileging of technical knowledge and left the Jamaican members with little room to influence the development process. Indian developers felt under significant pressure to complete their deliverables on time and paid little attention to the knowledge Jamaican MIS workers might contribute. At the same time, the critical sharing of technical knowledge by the Indians stopped as a Jamaican developer explained:

> The attitude changed to the Indians not assisting or sharing their skills and knowledge . . . Bad blood developed between Jamaicans and Indians.

As a result the tacit integration necessary to develop GENSYS effectively could not occur, as the Indians and Jamaicans became more isolated within their own occupational communities, and the boundary bridging activity of the spec ceased.

The Jamaican MIS staff did not adapt well to these unfamiliar software house routines of tight deadlines and 'coding to the spec', and felt their knowledgeability was undervalued and largely ignored. Instead of being able to contribute their tacit knowing of user expectations and contextual insurance knowledge to the software team this had been rendered subsidiary and they felt dominated and without a voice. A Jamaican developer noted:

> The feeling by most GENSURE staff was that the Indians had been given power over the Jamaicans . . . the whole project had been taken away from them.

While social boundaries and awareness of cross-cultural differences had always existed on the team, they became increasingly salient. The Jamaicans now perceived

Dr Prava's 'scientific, precise, and detailed project management' approach to be autocratic in 'laying down the law' and reflecting the Indian caste system. Jamaican workers believed the leader's monitoring and control of activities to be inappropriate, but Dr Prava thought this reflected cross-cultural differences towards hierarchy and control in the workplace:

> Everybody (Jamaicans) treats everyone as equal . . . if something is due at the end of the month you are not to intervene as the boss . . . The attitude is 'I will tell you if the job is done or not done' . . . they don't want a monitoring system . . . it is demeaning for the boss to ask about progress of activities in between tasks.

This contrasted with his experience in India as he went on to say:

> If I assign a job in India, as the boss I would ask if there were any problems at the end of the day . . . The Indian would not feel he is being watched but rather that I am helping him reach his end point.

The subsequent dismissal of the most technically competent Jamaican team leader, the Assistant Manager of the GENSURE MIS department, fuelled the climate of mistrust. The CEO of GENSURE explained:

> There was an atmosphere of mistrust and resentment, which resulted in a lack of ownership and political tensions . . . the culture differences gave [way to] mistrust and a blame culture ensued.

Labelling and stereotyping of Indians against Jamaicans as 'us and them' gave rise to a widespread belief that national culture was responsible for irreconcilable differences on the team. For example, a Jamaican team member commented, 'It is hard to relate to their caste system, where hierarchy and status were so important.' In contrast, the Indians felt that the Jamaicans avoided project coordination, and that Jamaicans were unable to 'link hands and do parallel work'. The Indians illustrated this through an example of Jamaican athletic performances where 'they are fantastic runners [but] they miss out on medals . . . because . . . the baton is dropped (though not the only ones to do so) . . . there is no training to coordinate'.

A few Jamaicans begged to differ with the predominant view that cross-cultural differences were to blame for the breakdown in trust and knowledge sharing between the groups. For example, the Finance Director, who initiated the insourcing arrangement, believed the key challenge was turf protection by the MIS group:

> The opposition by GENSURE MIS staff was not so much cross-cultural (as commonly argued) but a protection of turf.

The discontentment dramatically slowed the software development effort, and Dr Prava called a grievance meeting for demotivated team members to air their concerns. At the meeting, however, Dr Prava did little but reinforce the deep feelings of resentment the MIS staff felt towards control and their lack of ownership in the project. One Jamaican team leader explained:

> At the grievance meeting it was largely felt that the top positions were held by Indians, with Jamaicans working for them.

Jamaican staff's feelings of discontentment surfaced and subsequently increased as the expected benefits of learning software skills from their Indian counterparts disappeared with increased formal control. The imposed use of temporal boundary objects and the spec by the Indian project manager and the differing views and foci held by Jamaican and Indian developers of it led to task conflict as to how software development should ensue. This led to changes in interaction and a shift in the use of the spec as a source of collaboration and in facilitating knowledge exchange. Instead, the lack of knowledge sharing around the spec was now perceived to reinforce differences between them. The strict timeframes served to privilege the readily specifiable technical knowledge held by programmers, a cultural resource that reinforced their dominant position on the team. The unequal distribution of power relations among the subgroups within the team led to resistance and disengagement by Jamaicans. Knowledge sharing as a conduit between these cross-cultural groups ceased over time, however, and this served to reinforce and problematize the social boundaries between them. As a consequence, collaboration was replaced by relational conflict with negative emotions being exhibited and explained by cross-cultural differences, and the system was never successfully developed and implemented by the team.

Discussion

Our study posed the following research question: how and why can the use of requirements specifications and project management tools by members of a cross-cultural software team facilitate both collaboration and conflict at different points in time? We found that there were key interacting elements that influenced the role and use of boundary objects within the team. In the early phases, an open work climate and interdependence existed with mutual recognition and valuing of the knowledge provided by both Jamaicans and Indians. These interacting elements allowed for a flexible use of timelines and requirements specification and facilitated knowledge sharing and collaboration in the team. In a subsequent phase, however, there was a heavy reliance on formal coordination and control mechanisms as the Indian project leader asserted his managerial authority through the operationalization of strict deadlines and by mandating the team to 'code to the spec'. The subsequent use of the spec in software development had adverse implications for interactions between Jamaican and Indian developers. There was a shift from interactional processes of tacit knowing across team members in knowledge sharing to a more individual level focus on explicit knowledge in coding from the spec. The inherent privileging of technical knowledge in this activity shifted team dynamics away from interdependence and mutual recognition of subgroups' knowledgeability. Concomitant pressure of deadlines as temporal boundary objects decreased attention on collaboration and expertise coordination, and had adverse implications for shared identity of the team.

The final phase of the project highlighted how the distribution of power and authority to Indian technical specialists and the subsequent negative stereotyping of national culture led to boundary object use not effectively bridging or supporting the diverse groups

who were now isolated within their social boundaries. This had adverse implications for knowledge sharing and had implications for relational conflict (Hinds and Mortensen, 2005). 'Us and them' boundaries emerged between members of the team (Metui, 2006), and increased the salience of national culture (Randal, 2003). The indirect consequence of imposed and rigid boundary object use (e.g. spec) was therefore related to the detrimental impact of task and relational conflict on software development teams (Barki and Hartwick, 2001).

Our theoretical contributions build on a conceptualization of boundary objects as both pluralist, recognizing the potential for collaboration and conflict, as well as interactional; appreciating the interaction between boundary objects and their ongoing interpretation and use by different actors (Levina, 2005; Thomas et al., 2008). Rather than focusing on the objects themselves we take a broader view of the negotiations around the boundary objects as they are used in practice (Harvey and Chrisman, 1998). This situated and emergent perspective of boundary objects foregrounds the changing dynamics and how their interpretive flexibility is operationalized in practice, and pays attention to the wider interacting influences. We also demonstrate how Polanyi's concept of tacit knowing can be useful in understanding the challenges faced by actors using boundary objects to support sharing across knowledge boundaries. Tacit knowing inherently supports an interactional approach in developing meaning by building tacit and explicit components of knowledge through tacit integration of clues. It goes beyond recognizing that object's meaning is changing in use (Levina, 2005; Thomas et al., 2008) to further our understanding as to how meaning is tacitly derived across knowledge boundaries, and the role of boundary objects in enabling knowledge sharing. For example, using the spec as a boundary object allowed for co-construction and collective meaning to be negotiated among team members in the collaborative phase of ISD. Tacit knowing entails actively integrating clues drawn from focal and subsidiary objects that were interacting within numerous elements of practice. At the mid-point transition, the interacting elements of managerial authority and control shifted the clues that were focal such that the timelines were given more explicit attention as were explicit spec requirements. Users' needs and domain knowledge became subsidiary, particularly in the mind of the Indian programmers. Furthermore, this use of the spec did not allow Indian and Jamaican programmers to draw on knowledge from the other, limiting their potential to tacitly integrate meaningfully cues and clues from the spec for effective coding.

Our findings contribute to our understanding of the role and use of spec and project management tools as boundary objects particularly in relation to the political dynamics associated with pragmatic boundaries. Wastell (1999) has shown that requirements specifications and project management tools are important in supporting learning within ISD teams. Our interactional approach highlights how the same boundary objects can be used to both support and limit sharing across knowledge boundaries over time, and offers particular insights for how boundary object use may or may not overcome pragmatic boundaries in cross-cultural teams. In the early stages the boundary object use overcame vested interests that existed between groups as evident in the bridging and collaboration within the team. Later on, however, political tensions came to the fore as Jamaican developers felt dominated and without voice. Cross-cultural differences emerged as salient and became embroiled with vested interests and negative emotions within each group.

This triggered task and relational conflict and led to a situation whereby the boundary object no longer dealt with the pragmatic boundaries in the team, but rather reified the cultural differences across the group. We conceptualize this interacting element as *culturizing* and note the adverse effect it had on boundary object use in effectively overcoming pragmatic boundaries.

We contribute an understanding of how boundaries can lead to team conflict which becomes reified as being owing to 'cross-cultural differences' over time. While cultural differences were present at the outset, this did not result in conflict as knowledge sharing led to successful collaboration reinforcing salience of the team-wide boundary (Wimmer, 2008). Knowledge sharing and learning developed ownership (Metiu, 2006) and time was given to allow members to negotiate around the spec (Walz et al., 1993) and tacitly integrate clues to write meaningful code. However, this changed at the mid-point transition where the project manager drew on his authority and vertical power relations to redirect the use of boundary objects (Thomas et al., 2008) to regain managerial control. The role of the Jamaican group diminished and their tacit knowledgeability was poorly integrated into the coding process. This led to disengagement by Jamaican developers and a lack of ownership (Metiu, 2006) as their ability to exercise power in the co-construction of meaning was marginalized. With technical knowledge and skills centre stage, Jamaicans' contextual and domain knowledge was devalued, and the different types of knowledge reinforced symbolic boundaries between groups (Sherman, 2005; Vallas, 2001). Despite having domain knowledge, Jamaican developers were unable to alleviate consequences of status differences (Levina and Vaast, 2008) because top management valued the use of project management tools; the concrete and visual timelines provided them with stability over the inherent uncertainty of ISD (Wastell, 1999; Yakura, 2002). The operationalization of stringent deadlines and 'coding from the spec' as boundary objects reshaped the interaction between cross-cultural team members. The boundaries between Jamaicans and Indians became increasingly salient and minimal knowledge sharing occurred across this boundary.

Task conflict (Jehn et al., 1999) arose as Jamaican developers perceived little ability to exercise power in negotiating how the spec would be used and in operationalizing the timelines. They queried whether the task was primarily to complete coding on time or to develop an implementable program. The conflict around task goals related to how boundary objects were to be used. As the Indian leader dictated the timelines and spec be used in a way compatible with Indian software houses, issues about authority and control in the team came to the fore and made culture increasingly salient (Wimmer, 2008). This heightened symbolic and status differences between team members (Lamont and Molnar, 2002; Levina and Vaast, 2008) became reified and over time led to relational conflict, and further reinforced the salience of culture (Wimmer, 2008). The Jamaicans' lack of negotiability and identification manifested symbolic boundaries expressed as 'us' and 'them', which were fuelled by derogatory stereotyping (Dekker et al., 2003; Tajfel, 1981), a process we call *culturizing*. The use of the spec served to accentuate social boundaries as a result of occupational groups' level and type of knowledge (Vallas, 2001). This heightened status differences and feelings of exclusion (Metiu, 2006), which led to relational conflict (Barki and Hartwick, 2001), manifest on a blame culture based on cultural differences with respect to time, hierarchy and control, and differing mind sets (Vallaster, 2005).

Our findings on co-located cross-cultural teams add to the earlier team development literature (Gersick, 1988; Tuckman, 1965) as well as recent developments on virtual teams (Sarker and Sahay, 2002). As our team developed, initiation or forming was closely connected to, and quite quickly followed by, a collaboration phase, which was the stage in the groups' development that was most productive and hence 'performing'. Compared with virtual teams (Sarker and Sahay, 2002), this happened more quickly in our collocated team, seemingly as a recognition of interdependence and symmetrical valuing of knowledge between team members. The culturizing phase that followed the mid-point transition had some similarities with a storming phase (Tuckman, 1965) but was distinctive in clarifying the crucial importance of boundary object use within cross-cultural team development that led to conflict. Contrary to Tuckman's more positive and gradual transition stage model, which suggests that norming would follow storming in the development of cohesiveness and open exchange, we found that culturizing allowed for, if anything, a negative norming as a consequence of relational conflict and revealed a number of interactional elements that acted as triggers for transition (or not) from the current phase (Sarker and Sahay, 2002). Furthermore, our study supported the importance of Gersick's (1988) mid-point transitions in the software team development. A unique and distinctive element of our study is its cross-cultural dimension with a particular focus on boundary object use, which sheds insight as to the effects of transition between stages of team development that triggered conflict as a key interactional element. These insights have practical implications for managers of cross-cultural software development teams in understanding how different interactional elements, such as increasing managerial authority and control mechanisms, affect boundary object use and how subsequent culturizing around the mid-point transition can influence team development. Our article highlights that conflict affected team performance as evidenced by poor efficiency with significant project overruns, poor quality as demonstrated by the rewriting of code, and poor team viability with the groups entrenched in 'us' and 'them' positions for the duration of the project.

Conclusion

In our increasingly globalized world, the need to gain a richer understanding of knowledge sharing in culturally diverse software teams will continue to be of critical importance (Leidner and Hayworth, 2006). This article examines the changing role and use of boundary objects, which can trigger cultural clashes and interpersonal conflicts. We show how the role and use of boundary objects facilitated (or not) knowledge sharing between members. There can be different dynamics with the same objects facilitating collaboration at one time and reinforcing boundaries at other times.

We suggest that this approach is useful in understanding software development in cross-cultural teams. Though cultural differences undoubtedly are important aspects of team dynamics, our analysis warns of potentially undue singular attention to the politicization of cultural boundaries. Our plural and interaction approach to boundary objects allowed us to go beyond a focus on 'differences' to understand the slow down and eventual breakdown in knowledge sharing and coordination of expertise. In cross-cultural teams, this may accentuate the status differences of culturally diverse groups (Levina and Vaast,

2008), which may subsequently lead to conflict and negative stereotyping between groups. The implications of these findings are to unpack cultural explanations of poor ISD team performance by examining the processes underlying knowledge sharing associated with both the changing nature and use of boundary objects and how boundaries between cultural groups are made salient. To our knowledge this is the only case examining conflict in a South–South cross-cultural ISD team, though with the growing Indian economy and highly successful software market this is becoming more prevalent.

We close by acknowledging the limitations of our research, which has implications for the nature and extent of its generalizability. Our study included a small sample of teams consisting of participants from two companies within the financial conglomerate, and who were from only two national cultures. As such, the focus should be on analytical generalizability with future researchers drawing primarily on our theoretical approach to boundary object use, concepts such as culturizing, and implications of our findings in exploring knowledge sharing in other cross-cultural work teams. As such, we believe these developments offer a useful starting point for further research on boundary object use and knowledge sharing in a cross-cultural context.

Acknowledgements

We thank the Monieson Centre, Queen's University, Canada for its support of this research. We appreciated the feedback given by three anonymous reviewers and Paul Edwards, Associate Editor. We also thank the ORTEG group which provided feedback on an earlier version of this paper at Imperial College Business School. We also benefited from feedback given at the AOM international PDW, in particular from Lynne Markus, Samer Faraj, and Carsten Osterlund. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Note

1 The names of companies and individuals have been changed to maintain their anonymity.

References

Ayub N, Jehn K (2006) National diversity and conflict in multinational workgroups: The moderating effect of nationalism. *International Journal of Conflict Management* 17(3): 181–202.

Barki H, Hardwick J (1994) Measuring user acceptance, user involvement, and user attitude. *MIS Quarterly* 18(1): 59–79.

Barki H, Hartwick J (2001) Interpersonal conflict and its management in information system development. *MIS Quarterly* 25(2): 195–228.

Barrett M, Drummond D, and Sahay S (1996) Exploring the impact of cross-cultural differences in international software development teams: Indian expatriates in Jamaica. In: Coelho JD, Konig W, Krcmar H, O'Callaghan R, Saaksjarvi M (eds) *Proceedings of the Fourth European Conference on Information Systems*, Lisbon, Portugal.

Bechky B (2003) Object lessons: Workplace artifacts as representations of occupational jurisdiction. *American Journal of Sociology* 109(3): 720–752.

Briers M, Chua W (2001) The role of actor-networks and boundary objects in management accounting change: A field study of an implementation of activity-based costing. *Accounting, Organizations and Society* 26(3): 237–269.

Carlile P (2002) A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization Science* 13(4): 442–455.

Carlile P (2004) Transferring, translating and transforming: An integrative framework for managing knowledge across boundaries. *Organization Science* 15(5): 555–568.

Cohen C, Birkin S, Garfield M, and Webb H (2004) Managing conflict in software testing. *Communications of the ACM* 47(1): 76–81.

Cummings J (2004) Work groups, structural diversity, and knowledge sharing. *Management Science* 50(3): 352–364.

D'Adderio L (2004) *Inside the Virtual Product: How Organizations Create Knowledge through Software*. Cheltenham: Edward Elgar.

Dekker H, Malova D, and Hoogendoorn S (2003) Nationalism and its explanations. *Political Psychology* 24(2): 345–376.

DeSanctis G, Jackson B (1994) Coordination of information technology management: Team based structures and computer based communications systems. *Journal of Management Information Systems* 10(4): 85–110.

Eisenhardt K (1989) Building theories from case study research. *Academy of Management Review* 14(4): 532–550.

Ewenstein B, Whyte J (forthcoming) Knowledge practices in design; the role of visual representations as 'epistemic objects'. *Organization Studies*.

Faraj S, Sproull L (2000) Coordinating expertise in software development teams. *Management Science* 46(12): 1554–1568.

Ford D, Chan Y (2003) Knowledge sharing in a multi-cultural setting: A case. *Knowledge Management Research and Practice* 1(1): 11–27.

Gersick C (1988) Time and transformation in work teams: Toward a new model of group development. *Academy of Management Journal* 31(1): 9–41.

Harvey F, Chrisman N (1998) Boundary objects and the social construction of GIS technology. *Environment and Planning* 30(9): 1683–1694.

Henderson K (1991) Flexible sketches and inflexible data bases: Visual communication, conscription devices, and boundary objects in design engineering. *Science, Technology & Human Values* 16(4): 448–473.

Hinds PJ, Mortensen M (2005) Understanding conflict in geographically distributed teams: The moderating effects of shared identity, shared context, and spontaneous communication. *Organization Science* 16(3): 290–307.

Holden N (2001) Knowledge management: Raising the spectre of the cross-cultural dimension. *Knowledge and Process Management* 8(3): 155–163.

Jehn K, Mannix E (2001) The dynamic nature of conflict: A longitudinal study of intergroup conflict and group performance. *Academy of Management Journal* 44(2): 238–251.

Jehn K, Northcraft G, and Neale M (1999) Why difference makes a difference: A field study of diversity, conflict and performance in workgroups. *Administrative Science Quarterly* 44(4): 741–763.

Korsgaard A, Jeong S, and Mahony D. (2008) Multilevel view of intragroup conflict. *Journal of Management* 34(6): 1222–1252.

Kraut R, Streeter L (1995) Coordination in software development. *Communication of the ACM* 38(3): 69–81.

Lamont M, Molnar V (2002) The study of boundaries in the social sciences. *Annual Review of Sociology* 28(1): 167–195.

Langley A (1999) Strategies for theorizing from process data. *Academy of Management Review* 24(4): 691–710.

Leidner DE, Kayworth T (2006) Review: A review of culture in information systems research: Toward a theory of information technology conflict. *MIS Quarterly* 30(2): 357–399.

Levina N (2005) Collaborating on multiparty information systems development projects: a collective reflection in action view. *Information Systems Research* 16(2): 109–130.

Levina N, Vaast E (2005) The emergence of boundary spanning competence in practice: Implications for implementation and use. *MIS Quarterly* 29(2): 335–36.

Levina N, Vaast E (2006) Turning a community into a market: A practice perspective on IT use in boundary-spanning. *Journal of Management Information Systems* 22(4): 13–38.

Levina N, Vaast E (2008) Innovating or doing as told? Status differences and overlapping boundaries in offshore collaboration. *MIS Quarterly* 32(2): 307–332.

Lindkvist L, Soderlund J, and Tell F (1998) Managing product development projects: On the significance of fountains and deadlines. *Organization Studies* 19(6): 931–951.

Lyytinen K, Robey D (1999) Learning failure in information systems development. *Information Systems Journal* 9(2): 85–102.

Metiu A (2006) Owning the code: Status closure in distributed groups. *Organization Science* 17(4): 418–435.

Peltokorpi V (2006) Knowledge sharing in a cross cultural context: Nordic expatriates in Japan. *Knowledge Management Research & Practice* 4(2): 138–148.

Polanyi M (1968) *Knowing and Being*. London: Routledge & Kegan Paul.

Randal A (2003) The salience of culture in multinational teams and its relation to team citizenship behaviour. *International Journal of Cross Cultural Management* 3(1): 27–44.

Robey D, Farrow D, and Franz C (1989) Group process and conflict in systems development. *Management Science* 35(10): 1172–1191.

Sapsed J, Salter A (2004) Postcards from the edge: Local communities global programs and boundary objects. *Organization Studies* 25(9): 1515–1534.

Sarker S, Sahay S (2002) Understanding virtual team development: An interpretive study. *Journal of Association for Information Systems* 4(1): 1–38.

Sherman R (2005) Producing a superior self: Strategic comparison and symbolic boundaries among luxury hotel workers. *Ethnography* 6(2): 131–158.

Star S (1989) The structure of ill-structured solutions: Boundary objects and heterogeneous distributed problem solving. In: Huhns M, Gasser L (eds) *Readings in Distributed Artificial Intelligence*. Menlo Park, CA: Morgan Kaufman, 37–54.

Star S, Greismer J (1989) Institutional ecology, 'translations' and boundary objects: Amateurs and professionals in Berkley's Museum of Vertebrate Zoology. *Social Studies of Science* 19(3): 387–420.

Swan J, Bresnen M, Newell S, and Robertson M (2007) The object of knowledge: The role of objects in biomedical innovation. *Human Relations* 60(12): 1809–1837.

Tajfel H (1981) *Human Groups and Social Categories: Studies in Social Psychology*. Cambridge: Cambridge University Press.

Thomas R, Sargent L, and Hardy C (2008) Power and participation in the production of boundary objects. Working Paper Series, Cardiff Business School.

Tiwana A, Mclean E (2005) Expertise integration and creativity in information systems development. *Journal of Management Information Systems* 22(1): 13–43.

Tsoukas H, Chia R (2002) On organizational becoming: Rethinking organizational change. *Organization Science* 13(5): 567–582.

Tuckman B (1965) Development sequence in small groups. *Psychological Bulletin* 63(6): 384–399.

Vallas S (2001) Symbolic boundaries and the re-division of labor: Engineers, workers and the restructuring of factory life. *Research in Social Stratification and Mobility* 18(1): 3–39.

Vallaster C (2005) Cultural diversity and its impact on social interactive processes. *International Journal of Cross Cultural Management* 5(2): 139–163.

Vaast E, Levina N (2006) Multiple faces of codification: Organizational redesign in an IT organization. *Organization Science* 17(2): 190–201.

Walsham G (2002) Cross-cultural software production and use: A structurational analysis. *MIS Quarterly* 26(4): 359–380.

Walsham G, Sahay S (1999) GIS for district-level administration in India: Problems and opportunities. *MIS Quarterly* 23(1): 39–66.

Walz D, Elam J, and Curtis B (1993) Inside a software design team: Knowledge acquisition, sharing and integration. *Communications of the ACM* 36(10): 63–77.

Wastell DG (1999) Learning dysfunctions in information systems development: Overcoming the social defenses with transitional objects. *MIS Quarterly* 23(4): 581–600.

Wimmer A (2008) The making and unmaking of ethnic boundaries: A Multilevel process theory. *American Journal of Sociology* 113(4): 970–1022.

Yakura E (2002) Charting time: Timelines as temporal boundary objects. *Academy of Management Journal* 45(5): 956–970.

Michael Barrett is Director of Programmes and Reader in IT and Innovation at Judge Business School, University of Cambridge. He is co-lead on the National Institute for Health Research (NIHR) grant in Collaborations for Leadership in Applied Health Research and Care. His research interests include service innovation, open innovation, cross-cultural teams, and knowledge translation. He has studied the insurance sector, electronic trading in global financial markets, global software outsourcing, and service delivery within healthcare. His work is of an interdisciplinary nature, with publications in a wide range of journals including *Academy of Management Journal*, *Information Systems Research* and *Accounting Organisations and Society*. He has recently co-edited a book on IT in the service economy. [Email: m.barrett@jbs.cam.ac.uk]

Eivor Oborn recently earned her PhD from Judge Business School, University of Cambridge and is currently Lecturer in Public Management and Organization at Royal Holloway University of London. She is also co-lead on the NIHR grant in Collaborations for Leadership in Applied Health Research and Care. Her research interests include multidisciplinary teams, knowledge translation, technology adoption, and use and organizational change. She has papers accepted for publication in *British Management Journal*, *Public Administration* and *Journal of Health Service Research and Policy*. [Email: eivor.oborn@rhul.ac.uk]