The role of project management in ineffective decision making within Agile software development projects

John McAvoy[1] and
Tom Butler[1]

[1]Business Information Systems, University College Cork, Ireland

Correspondence: John McAvoy, Business Information Systems, University College Cork, Ireland.
Tel: +353 21 490 3338;
Fax: +353 21 427 1566;
E-mail: j.mcavoy@ucc.ie

Abstract

Decision making in traditional software development lies with the project manager. In contrast, Agile software development teams are empowered to make decisions, while the role of project manager has changed from one of command and control (i.e. to make decisions and ensure they are implemented) to one of a facilitator. This article argues that decision making in software development is not characterised by a sequence of isolated or exclusive decisions; rather, decisions are inter-related, with each decision leading to further decisions, the chain of which often spans the entire duration of a project. Over this extended period, there are several potential factors that can negatively affect the efficacy of decision making by Agile teams. One of the findings of this exploratory longitudinal study is that the high level of empowerment of a cohesive software development team undertaking an Agile project may be one of these negative factors, as empowered, cohesive teams can exhibit problems such as groupthink or the Abilene Paradox. This article therefore argues that the role of project manager in Agile development initiatives needs to be reassessed, with project managers taking on the role of devil's advocate in the decision-making process.
European Journal of Information Systems (2009) **18,** 372–383.
doi:10.1057/ejis.2009.22; published online 28 July 2009

Keywords: Agile; project management; empowerment; decision making; devil's advocate

Introduction

Agile software development (ASD) is now widely used in the software development industry; accordingly, it has been the growing focus of research. The decision by organisations and project teams to adopt an Agile methodology is of particular interest to researchers, and studies in this area have proposed techniques and tools to support decision making around Agile adoption (see, for examples, Boehm & Turner, 2003, 2004; McAvoy & Sammon 2006). Research on group behaviour from reference disciplines suggests that decisions to adopt and subsequently implement and use ASD may be much more complex than previously assumed; the motivation for this study is, therefore, to investigate this phenomenon.

Decision making is argued to be characterised by the following activities: (1) identifying the need for action; (2) deciding on which action to take; (3) determining the ends associated with such action; (4) committing to take that action; and (5) maintaining this commitment to the end (Mintzberg *et al.*, 1976). It is clear from Mintzberg *et al.* that decision making is longitudinal in nature; however, throughout the life cycle of

a decision circumstances and commitments may change, necessitating further decisions. Such decisions are typically related to questions like: Do we continue on as before? Do we need to alter the original decision? Do we need to reassess the actions decided upon? Thus, Robbins & Finley (1998) argue that the operationalisation of a decision is more important than the method adopted to arrive at the initial decision.

We contend that the decision to adopt a software development methodology aligns more with a longitudinal view of decision making than with conceptualisations of decision making as a once-off phenomenon. Furthermore, over the life cycle of a decision, several factors can influence decision making and related outcomes – group interaction, for example, is particularly influential in shaping decision making and outcomes (McGrath, 1984). Thus, it may be deduced that in a study of the adoption and use of an ASD by project managers and their teams, there needs be a consideration of decision-making processes beyond that of the original single decision point. Thus, the focus of such a study needs to be broadened to include the entire life cycle of a decision – that is, there is a need to investigate the inter-temporal influences on decision outcomes, to describe what actions have been taken, who has taken such actions, and why those particular actions have been taken (cf. Salo et al., 2004). Given that in ASD there is typically a change in the locus of decision making from the project manager to the software development team, the objective of this study is to investigate, understand, and explain the variety of influences on team-based decisions surrounding the adoption of an ASD approach.

The remainder of this article is structured as follows: The second section reviews the extant literature on ASD and presents theoretical insights that suggest concerns with decision making in an ASD team. The third section presents the study's research method, which employs participant observation to uncover the latent influences on team-based decisions in Agile projects. The fourth section presents theoretically informed descriptions of the phenomenon of interest in the two cases purposefully selected for study. The final section then offers conclusions and recommendations for both practice and future research.

Empowerment and project management in ASD projects

What makes ASD interesting and unique among IS development methodologies is its inherent philosophy on decision making by project teams; there is, however, a noticeable dearth of research on the socio-psychological forces that influence decisions taken by team members and on the outcomes of such decisions. In conducting research into this phenomenon it is important for researchers to note that there exists no single ASD methodology; rather, Agile is a collection of methodologies that have a core of common principles or values (Abrahamsson et al., 2002; Erickson et al., 2005). The

Agile manifesto (http://agilemanifesto.org) describes the core values that underlie all Agile methodologies viz.

- individuals are more important than processes and tools;
- working software is more important than comprehensive documentation;
- customer collaboration is more important than contract negotiation; and
- responding to change is more important than following a plan.

The first value – individuals are more important than processes and tools – is, we argue, the most relevant for decision making, as software development team members form the nucleus of actors in the ASD process. Highsmith (2004), for example, emphasises the importance of a good team for the success of ASD projects, while Hazzan & Tomayko (2003) describe eXtreme Programming (XP) – one of the most popular ASD methodologies – as being based on cohesive teams and their interaction (cf. Chin, 2003). The emphasis on team-based factors in Agile is therefore more important than in other software development methodologies, as ASD teams are generally cohesive and are empowered and expected to make day-to-day decisions. Thus, Schuh (2004) argues that empowerment and trust are central to ASD, as software teams are argued to have collective responsibility for the delivery of all functionality (cf. Cohn, 2004). However, for empowerment to work, researchers argue that teams need to be well-functioning and cohesive – Auer et al. (2003), for example, view ASD teams as effective social networks that interact well (Boehm & Turner, 2003), which are bound tightly together (Highsmith, 2004), and cohesive (Chin, 2003). It is important to note that this does not imply that the adoption of ASD will make a project team cohesive; rather, it means that a cohesive team is an *a priori* condition for Agile approaches to work.

Extant descriptions of ASD team characteristics marry well with conceptualisations of cohesiveness found in reference disciplines viz. it is defined as *the degree to which members of the group desire to remain in the group* (Cartwright & Zander, 1968; Hare, 1994; Kim, 2001, p. 175; Statt, 2004) and as interpersonal attraction, task commitment, and group pride (Beal et al., 2003). It may, therefore, be argued that ASD can be seen as more aligned with McGregor's (1960) Theory Y as opposed to Theory X approaches to managing software development teams. For example, Theory Y postulates that team members will be more effective when decision-making responsibilities are delegated to them (cf. Cartwright, 2002; Landy & Conte, 2004). In light of these points, the emphasis on empowerment and collective decision making in ASD projects necessitates a sensitive approach to project management in Agile projects. There is, however, a paucity of empirical research in the area of Agile project management (Abrahamsson et al., 2002). Significantly, Roock & Wolf (2004) state that project control was not originally emphasised by the Agile movement – a fact that provides

further impetus for researchers to undertake research in this area. This article contributes to this body of research through its investigation of the phenomenon.

Augustine (2005) defines Agile project management in terms of the empowerment of the team to rapidly deliver value to a customer or customers. In this scheme of things, Agile project managers provide a vision of what is required to their teams and the teams decide how to fill in the details (cf. McBreen, 2002). Furthermore, unlike traditional methodologies, Agile project managers do not operate a conventional command and control structure. Augustine (2005) argues that command and control is kept to a 'barely sufficient' minimum, as collaboration and self-management replaces command and control in Agile teams. Participatory decision making is an example of this concept of 'barely sufficient' command and control. In traditional projects, project managers make the decisions, while in projects where participatory decision making occurs, the views of the entire team inform decision making. Highsmith (2004), therefore, argues that participatory decision making is a core characteristic of ASD teams. This approach does not mean an abdication of responsibility by a project manager though, as on occasion he/she may have to make an overriding decision, rather than agree with a sub-optimal decision arrived at through team consensus. Nevertheless, project managers having the last word is deemed acceptable in Agile projects provided all opinions have been taken into account. Thus, Augustine (2005) describes the role of project manager in participatory decision making as one of facilitating and working with a team in making project-related decisions. This approach is neatly summarised by Schuh (2004, p. 164), who states that 'the Agile project manager acts more like a facilitator and less like a foreman. Instead of telling each programmer what to do and how much time there is to do it, the Agile manager fosters an environment where members of a team are able to make decisions and base them on the best information available'.

It is characteristic of many teams (be they software development teams or not) that they are empowered to make what are group decisions; that said, what is unique about Agile teams is the level of empowerment bestowed upon them (Stephens & Rosenberg, 2003). Thus, Stephens and Rosenberg (*ibid*., p. 94) point out that 'Agile methods have a much higher emphasis on people then previous methodologies'. Chin (2003, p. 87) goes further by arguing that 'a cohesive team is, very possibly, the difference between success and failure in the Agile environment'. As previously outlined, in traditional command and control project management scenarios, decisions are implemented by the team, with decision-making responsibility and ownership being conferred on the project manager. To reiterate, an ASD team makes and is responsible for project-related decisions; for example, McBreen (2002) describes how in XP, software developers take decisions on particular courses of action, provided they still meet project's goals. Of course, taking this approach to its

logical conclusion (or beginning), participative decision making should begin with the decision to adopt an ASD methodology by an existing software development team, and should continue from the initial decision to adopt through to subsequent decisions to continue as originally planned or change the original course of action.

Problems with decision making in ASD
As indicated above, the decision to adopt and implement an ASD approach involves a network of decisions over an extended period – which may span the life of a software development project. Thus, the initial decision to adopt ASD is merely the first of many decisions by a software development team. Hence, it is argued here that an empowered and cohesive ASD team, as opposed to just the project manager, will influence these decisions and, therefore, the ultimate outcomes associated with the initial decision to adopt Agile.

Group cohesion is generally regarded as having a positive influence on team performance. Basadur (1997) and Balthazard *et al.* (2004), for example, found that cohesion increased a group's overall performance. Similarly, Thamhain & Wilemon (1997) argue that good team spirit and trust, which are good indicators of team cohesion, are requirements for high performance teams. High levels of team spirit (Von Bergen & Kirk, 1978) and trust (Langfred, 2004) are associated with highly cohesive teams. In effect, Thamhain & Wilemon (1997) correlate high levels of team cohesion with high performance teams. Despite these benefits, cohesion has been indicated as a source of ineffective or dysfunctional decision making. Perhaps the most noted (and notorious) problems associated with team cohesion is groupthink (Janis, 1972), while a related concept, the Abilene Paradox (Harvey, 1974), is closely associated with groupthink.

Groupthink is defined by Janis (1972, p. 9) as 'a deterioration of mental efficiency, reality testing, and moral judgment that results from in-group pressures.' Ottaviani & Sorensen (2001) define it as the psychological drive for consensus at any cost that suppresses disagreement and prevents the appraisal of alternatives by cohesive decision-making groups. Levine & Moreland (1990) do not simply use the term consensus, they describe groupthink as extreme concurrence seeking, which Manz & Neck (1995) and Hogg & Hains (1998) argue leads to dysfunctional decision making. The ultimate deciding factor in the occurrence of groupthink is, paradoxically, a highly cohesive team (Janis, 1972; Manz & Neck, 1995; Hogg & Hains, 1998; Kim, 2001). The view of the group becomes the view of the individual, and the individual will accept the group's views as being correct (Janis, 1972; Manz & Sims, 1987). Ultimately, the individual will happily profess the group's view as their own (Kim, 2001). While Gowda & Chand (1993) include person to leader attraction in their definition of group cohesiveness, Janis adds that groupthink does not imply either a dictatorial leader or sycophantic followers. Leaders may genuinely

believe that they are open to, and look for, honest opinions. Similarly, group members may not be afraid to express their opinions. However, what prevents true discussion and disagreements are subtle socio-psychological factors and constraints that prevent dissent or departures from group norms and values.

A similar cause of ineffective decision making is referred to as the Abilene Paradox, which bears similarities to groupthink and, as such, may be influential in shaping decision making in the adoption of an ASD. Harvey (1974, p. 65) describe the Abilene Paradox as 'the inability to manage agreement' (cf. Thomas, 1990). The name of this concept originates in a group decision by the extended family of Professor Jerry Harvey to take a day trip to Abilene. The paradoxical nature of the decision to take the journey to Abilene and the recriminations that followed are described in several articles (cf. Harvey, 1974; Harvey, 2001; Robbins & Finley, 1998; Kim, 2001). On a hot day in July, members of the extended Harvey family were sitting contently on the back porch of Harvey's father-in-law's house. The origins of the visit to Abilene stemmed from a suggestion by his father-in-law, who was looking for somewhere for the family to eat out. Harvey recounted that at the time he thought it a bad idea under the circumstances, but decided to keep his views to himself. All agreed to take the trip, which turned out to be less than ideal; in fact it was a disaster. On their return from Abilene, the recriminations started and all family members confessed to thinking that that it was a bad idea when it was first proposed and had not wanted to go to Abilene. Paradoxically, family members blamed each other for taking the decision to make the journey. Even the person who suggested the excursion, Harvey's father-in-law, confessed that he did not want to take the trip, but proposed the idea as he thought other family members might like it. This peculiar form of dysfunctional decision making Harvey (1974) termed the Abilene Paradox; he conceptualised this as a form of collective decision making where a group decides on a course of action that no single member would have taken if they were the decision maker.

A description of a variant of groupthink and the Abilene Paradox comes from Patching (1999), who argues that although individuals in a group may accept a decision and intend to implement it, subconscious beliefs and values act to prevent the decision from been carried out successfully. Patching illustrates that the desire for cohesion, coupled with a reluctance to express views contrary to what is assumed to be the group's view, leads to symptoms of ineffective decision making such as groupthink or the Abilene Paradox.

Drawing on the findings and arguments made in previous research, this study assumes that effective decision making by Agile teams is a function of the level of cohesion existing in a team and the subsequent level of empowerment in decision making bestowed upon the team by the project manager. The article also assumes that the desire for continued team cohesion, coupled with reluctance by team members to express views contrary to what are assumed to be views of other team members, and not just the project manager, leads to dysfunctional and ineffective decision making, such as groupthink or the Abilene Paradox.

Research approach
Given the above assumptions, this study's objective is to examine how decision making was influenced in two ASD teams over the life cycle of the initial, collective decision to adopt this approach to software development.

The underlying causes of problems with decision making in organisations, teams, and with individuals such as project managers, are often latent or hidden, even from those involved in the process. For a variety of social and psychological reasons, social actors are usually unable to determine the causes of such problems and why they occur. This situation calls for the application of research approaches that are sensitive to such issues (Jorgensen, 1989). Thus, to investigate problems with decision making in software development teams necessitates a longitudinal, exploratory, research approach that permits the team to be examined in context. Furthermore, the investigation of the social factors that inhibit decision making involves the examination of phenomena that would be normally hidden from, or not observable by, outsiders; indeed, individual team members themselves may not be aware of the existence of such influences (Goleman, 1996). Argyris (1976) illustrates how individuals are unable to discern the difference between what they believe in and what they actually do (espoused theories *vs* theories in use) and it is clear from Argyris that phenomena such as groupthink and the Abilene Paradox are usually hidden from those affected by the phenomenon. Surveys, questionnaires, and other such approaches are therefore unsuitable research methods as a respondent cannot, or may not wish to, identify the differences between their espoused theories and their theories in use. To overcome these difficulties, a qualitative, case-based research approach utilising participant observation was adopted for the study (cf. Yin, 2003). A number of studies have employed participant observation to conduct research on ASD projects. For example, a qualitative approach involving participant observation was used by researchers to investigate the characteristics of an Agile team and provided rich insights that could not be obtained by other research methods (cf. Robinson & Sharp, 2005).

The overall research approach involved a multiple case study strategy (Yin, 2003). A purposeful, replication sampling logic was adopted (Patton, 1990; Yin, 2003) in order to select two software development project teams that had decided to introduce an ASD approach. Both project teams had already designed and developed information systems for use by a variety of organisations. The two teams (both of which comprised six developers and a project manager) were about to employ ASD

approaches in software development projects that would last approximately 1 year each.

As indicated, participant observation was chosen as the primary research technique to investigate the phenomenon of interest, as it is a particularly relevant approach when 'the phenomenon is obscured from the view of outsiders' (Jorgensen, 1989, p. 12). Participant observation of both development teams occurred at regular intervals over a period of 1 year. The principal investigator participated in both teams and was, therefore, able to take part in team meetings, formal and informal discussions, and so on. Such activities are argued to be vital in participant observation (Ezey, 2003), as it 'allows you to experience activities directly to get a feel of what events are like, and to record your own perceptions' (Spradley, 1980, p. 51). Rather than being an outsider looking in, the researcher was an insider working with the team while researching the phenomena – he was, as Bødker & Pedersen (1991) put it, a cultural insider. Detailed field notes were taken throughout the research process and these were reflexively analysed and recorded by the researchers. A wealth of documentary evidence was also gathered. All project documentation was made available to the researcher, from process documents to performance statistics. While observation was the primary data source, supplemental and confirmatory information was acquired through documents and interviews. The theories presented herein acted as a lens in the interpretation and analysis of the data, which began at the point of collection. The various themes, issues, and group interactions were identified as the study progressed and initial observations subsequently confirmed.

Following Yin (2003), the internal validity or credibility of the study was achieved through techniques such as prolonged engagement, participant observation, triangulation, peer debriefing, and member checks or venting. The venting technique described by Goetz & LeCompte (1984) merits particular mentions as it was employed for member checks in that the researchers' interpretations and findings were formally discussed with individual team members to ensure that the analysis and conclusions were taking all relevant factors into account. This approach also facilitated developers and project managers to provide further input and interpretations on events. The external validity or generalisability of the study was ensured through thick description, purposive sampling of the cases that permitted literal replication (Yin, 2003), and through the use of a reflexive journal/field notes; however, the technique of analytic generalisation of the findings of both cases to the study's 'rich theoretical framework' was of particular importance (Yin, 2003, p. 47). The issue of reliability and objectivity were addressed through triangulation, an audit of the research artifacts, and the reflexive journal/field notes.

Observations and analysis

The observations generated during the research process on both software development teams are presented below. The evidence provided illustrates that although the decision-making processes differed in each project, the outcomes of both were similar. For example, ineffective decision making characterised both cases in the adoption and continued use of the ASD approaches employed by the teams, as did the role played by the project managers in fostering such dysfunctional behaviours.

The Knowledge Management System development team

The first software development team studied was involved in the development of a Knowledge Management System (KMS) for use in a large government department. The design of this IS was informed by the experiences of the team in developing a KMS for the United Nations (UN). The majority of the team had worked together on the UN project and was highly rated by the customer. Their experiences on that project led to the team being a highly cohesive unit with developers working well together, whatever the task; another indicator of the close working relationships that developed between team members was that they socialised with each other outside of office hours. Also significant is that the project manager was well respected by the team, and his decisions were usually accepted during team discussions, which were generally relaxed, informal affairs. One of the reasons for this was that he empowered the team, and had a great degree of trust in its members. In contrast to traditional project management styles, his relaxed, hands-on approach saw him work as one of the team, taking part in technical decisions, more like a developer than a project manager. While he entrusted day-to-day decisions to the team, he was ready to step in to assist with decision making where appropriate.

It is significant that it was the project manager who first suggested the use of Agile as a software development approach. It was tight project deadlines and limited developer resources that had him suggest Agile as a mechanism to help the team deliver a solution for the customer rapidly and with the desired level of quality. However, it was merely a suggestion, and in keeping with his project management style, he left the decision to adopt Agile to the team, as they were going to be the ones using it. Drawing on their experiences in the previous project, team members felt that the structures provided by Agile would assist them in developing the government KMS as productively and as quickly as possible. The decision to adopt and use Agile was therefore a collective one. From the outset, the project management approach adopted was viewed by team members as one of empowerment, rather than the traditional command and control approach. Thus, from the outset, decisions as to the development methodology, tools (e.g. the integrated development environment used), and techniques (e.g. user stories etc.) were made in open discussion with the team, as opposed to the project manager

Table 1   Occurrence of groupthink in Knowledge Management team

Groupthink symptom	Occurrence in case study
Little or no consideration of alternate plans	The project manager and developers decided against the use of several Agile processes/techniques in favour of less effective *ad-hoc* traditional methods.
Risk is not assessed	The risks associated with not following the Agile approach, such as recommended methods and omitting certain techniques, were not assessed: the participating researcher pointed out several of the risks but was collectively ignored or dismissed by the project manager and the team.
No review is taken of rejected plans	When the participating researcher, with the backing of one of the developers, attempted to review decisions made to drop some of the methods, the project manager stated that 'we should all be singing from the same hymn sheet'; the team agreed with this and the discussion ended.
Advice from outsiders is not sought	When the participating researcher cited recommendations from Agile experts that supported the use of particular Agile practices which the team decided not to use, he was informed that they were 'not applicable in this case', that he was being extreme, and was going against the team's decisions.
Facts that support the plan are acknowledged, facts that do not support the plan are ignored	When it became clear that the decision to change the prioritisation method recommended by Agile approach was a bad decision, the correct method was not reconsidered for use.
Contingency plans are not created	The team had the utmost faith in their decisions and never examined alternatives. Their belief, their own decision-making capabilities to get things right was driven by the fact that the team had been very successful in a previous project.

imposing them, which would be the case with command and control approaches.

Once the initial decision to adopt Agile was made, subsequent decisions were made on the specifics of the adoption. While initial team members' attitudes were very supportive to the use of Agile, there was, over time, a gradual erosion of support for its use. The dilution of support for Agile in subsequent decisions to, for example, document user requirements, was not immediate, but occurred gradually after the first 2 months of the project, when team members became immersed in the day-to-day activities of the project. The first indication of what was a gradual erosion of support came during the decision to utilise User Stories (Agile's method of documenting requirements). The project manager supported their use, but argued for minor changes in how they would be used. Thus, an approach was used that was essentially a hybrid of traditional requirements gathering and Agile User Stories. This decision was discussed at relevant meetings by the team and unanimously supported; in support of this position, developers argued that the specifics of their project were different from projects described in the Agile literature.

Shortly after this decision was made, came the decision to prioritise the user requirements documented using User Stories. XP (the Agile method chosen) argues for a numerically ranked list of requirements. However, the project manager noted that the developers did not appear to be fully committed to this concept, as they felt it was much too specific and troublesome to implement. As a compromise, he recommended that requirements be simply ranked high, medium, and low. The team discussed the proposed compromise and unanimously

agreed with the project manager that this ranking system was a good concession. The project manager subsequently justified his recommendation by arguing that the developers knew best and the compromise approach would work effectively if the developers were given flexibility. Subsequently, the decision was seen to be suboptimal as practically all requirements were ranked in the high category, making it impossible for developers to differentiate between them. Rather than reviewing the original decision, all prioritisation ceased and developers made individual decisions on which requirements were going to be implemented and when.

The findings of this first case study highlighted several other examples of decisions such as this; however, on each occasion team unity and cohesion won out over good decision making. Thus, the overarching desire for conformity highlighted the possibility of groupthink influencing decisions and project outcomes. This was confirmed when the symptoms of groupthink were compared to observations during the study (derived from Janis (1972) and shown in Table 1).

While groupthink was seen to affect decision making by the team as a whole, it was apparent that decision making by the team was overly influenced by the project manager. Previous mention was made of his relaxed project management style, and hands-on approach to practical project-related matters and activities. It was in this latter context that his influence was inadvertently exercised. Interviews with the developers showed that there was a reluctance to go against the project manager (even though the project manager empowered them in the decision-making process). This reluctance to disagree with the project manager was based on respect for him as

a team member, rather than any negative fear of reprisal or criticism. Thus, rather than a reluctance to disagree out of fear of the consequences, or because of religious adherence to a hierarchical command and control structure, the team usually agreed with the project manager's recommendations out of respect, loyalty, and belief that he would take their considerations seriously in the decision-making process.

The type of groupthink observed in this team closely matches a variant of groupthink called hierarchical groupthink. Huczynski & Buchanan (1991) argue that synergy and loyalty to each other and to the team leader are a team's greatest qualities; however, Huczynski and Buchanan argue that they are the same factors that lead to groupthink. Hierarchical groupthink originates in a desire to please a leader through agreement with opinions or decisions. This phenomenon was also described by Neck (1996), who lists leader preference for a particular decision as a potential factor in groupthink among subordinates. Manz & Neck (1995) and Hogg & Hains (1998) also describe the desire to conform to a leader's views as groupthink, although the term hierarchical groupthink is not used by these researchers.

What is interesting about the observed behaviours of team members in the KMS project is that the project manager was genuinely unaware that his views were being regarded as sacrosanct by the team. He empowered the developers to make decisions and, while he was expressing his opinions, he regarded his part in the decision-making process as that of being just one of the team, rather than the ultimate decision maker.

Could this problem have occurred in a traditional development team? While similar problems may have occurred, the reasons behind them would differ. As indicated, groupthink developed in this team not due to fear of the project manager's authority. The developers agreed with him because he was an Agile manager; he was respected because he empowered and trusted his team. The developers felt that if he expressed an opinion then it was worthy of consideration, exactly because he did not force his views on the team. Thus, the project manager's opinions were accepted and agreed with by the team, as they trusted his judgement, as did he theirs. This empowerment therefore, while well meaning, was suboptimal. Although the manager believed in empowering the developers, the esteem he was held in meant that his opinions had a greater impact than they should have in a truly empowered team.

In a traditional project with a command and control view of project management, it could have been possible to coerce the team into accepting the project manager's view, but this would not have led to the same problem. With hierarchical groupthink, the developers agreed and supported the decisions, not because they had to, but because they believed (or at least convinced themselves) that they were correct. In fact, this made the groupthink worse as they ultimately believed and argued for the decisions themselves. With a non-Agile project manager,

the team may have accepted the decision, but there would have only been external commitment as opposed to internal commitment (as described by Argyris, 1998). Only when internal commitment is present can true hierarchical groupthink arise, as opposed to external commitment where the team is following edicts or instructions. In applying Agile principles to the project, the project manager created this internal commitment, which ultimately led to hierarchical groupthink and dysfunctional decision making.

The telecommunications projects team

The second team studied consisted of software developers involved in the development of an information system for use by customers of a global telecommunications equipment manufacturer. This team of six developers and a project manager was highly rated by company management. One senior manager, who had no direct responsibility for the software development team, stated that they were 'the best team in the company'. As with the other software team studied, this team of software developers were also a highly cohesive unit both professionally and socially. Group norming and performing over time had the team develop their own language and customs and, as with such familiar social groupings, shared several 'in-jokes' that were unique to the group and not accessible to outsiders. All this helped shape the team's identity and character. The project manager was well respected by the team and the developers regarded him as 'one of their own'. Collectively, the team distrusted other groups and managers; so much so, that they regarded them as a nuisance who threw obstacles in the team's way, and who had no idea what the team did.

The adoption of ASD for a new project was a collective team decision. The particular approach adopted by this team was a hybrid of XP and Dynamic Systems Development Method (DSDM). It is significant that initial comments by all team members, including the project manager, on the use of XP and its integration with DSDM were very positive. They all stated that the hybrid approach was simple and straightforward and a 'great idea' that would help them to deliver better software quicker and with fewer errors than their traditional approach.

The first phase of the adoption of Agile saw the team employ User Stories to document customer requirements. However, as this telecommunications organisation employs mandatory business methods and standards for software development processes, the team had to make modifications to the way in which the technique of User Stories was employed in order to adhere to company's quality standards. At this point, the project manager and developers were pleased with the modified User Stories method and they were of the opinion that Agile was well suited to their needs. Thus, the project manager empowered the developers to continue with the adoption of Agile methods.

After a brief honeymoon period, there was a noticeable decrease in the team's enthusiasm for the Agile approach.

Well before the mid-point in their year-long software development project, support gradually decreased to a level where Agile was no longer looked at in a positive light by either the project manager or his team. Interviews conducted at this point in time had the team allege that external organisational factors were negatively influencing the adoption and use of Agile, so much so, that working around issues was felt to be a 'waste of time'. Interestingly, the external factors identified by the developers and the project manager were associated with the activities of company's Quality Assurance Group. This was a strange development as from the outset the Quality Assurance Group had been very supportive in the adoption and modification of User Stories in order to gather and document requirements in line with company policy. Strangely, when asked for concrete examples of negative impacts, the developers and project manager made comments such as 'that's just what they are like' and 'there's no point if they get involved'. Follow-up interviews held with management and staff of the Quality Assurance Group disconfirmed this notion. The Quality Assurance Group stated that while they fully supported the use of Agile methods, there were certain procedures that had to be followed; nevertheless, it was pointed out that these procedures were both high-level and flexible and did not, as such, materially affect the adoption of Agile. It is significant that not one team member had discussed their concerns with the Quality Assurance Group. Hence, using feedback from the Quality Assurance Group, and from observations over the 6-month period in question, the participating researcher probed further into the cause of the failures during the life cycle of the decision to adopt Agile (i.e. implementation and use decisions subsequent to the initial adoption decision).

Each software developer on the team was interviewed individually and in confidence and asked as to why subsequent decisions (e.g. such as the failure to use software product iterations as per Agile's recommendations) went against the initial decision to adopt. When confronted with the evidence from the quality group that their policies were not impacting negatively on the adoption of Agile methods by the team, each of the software developers acknowledged that this was not, in fact, the real problem. It was surprising to find that all of the developers individually blamed the project manager for the failure to fully implement and use Agile. The developers were unanimous in stating that although they still supported the decision to adopt Agile, there was no point in arguing for its continued use if the project manager had negative feelings towards it. They also stated that there was nothing to be gained from arguing against the project manager's viewpoints at meetings in order to influence what were meant to be team decisions. Strange as this may seem, it was even stranger to discover that individual developers were of the opinion that their teammates did not feel as strongly or as positively as they did on the issue of Agile's adoption, so there

appeared to be no point in arguing against them also. This created a paradox, as each developer wanted to adopt Agile, yet felt that the other developers did not, when in fact all shared the same view. A further paradox arose in the interviews with the project manager, who indicated that he wanted to continue using Agile, but he felt that the team did not. Because he believed in empowering his team, he had to respect their decisions (even when such decisions were to discontinue the use of Agile); thus, he remained silent on the issue at meetings, or if he did speak, it was to support their perceived position. This meant that his empowerment of the team was suboptimal. Empowerment does not mean abdication of the right to, or necessity to, offer an opinion. The project manager in this instance believed empowerment to mean that his opinion had to be withheld, whereas true empowerment of the team should have allowed his opinion to be heard as one opinion within the team.

The words 'perceived position' is important here, as the perceptions of the software developers and the perceptions of the project manager were both incorrect. In what reads like a Shakespearean farce, the developers felt that the project manager was against the use of Agile, because he never expressed any positive opinions during its use. The developers did not believe that the project manager forced them into abandoning Agile (they all acknowledged that decision making was collaborative rather than hierarchical); rather they felt that 'he did not push the issue', to quote one developer. Thus, they took this to indicate a lack of support for the initiative by him. The project manager admitted that he did not express any positive opinions because he believed that the developers' decision was to drop the use of Agile. He stated that 'if the team don't want to do it, then I am not going to make them – that's not how we do things'. This almost comical situation suggests the existence of the Abilene Paradox (Harvey, 1974): the findings are now analysed with respect to Harvey's theory, and presented in Table 2.

The first symptom of the existence of the Abilene Paradox identified by Harvey (1974) is that group members know what their preference is or what is the correct decision to take, but this is not shared with the group. All developers on this project wanted to fully adopt the Agile approach, but would not push for the decision to be implemented fully. The second symptom is that group members hold similar perspectives on how a problem can be resolved, but they do not share these with the group. In the above case, each software developer stated in private how Agile should have been adopted, but did not share this with the rest of the team.

The third symptom is that instead of communicating their views, group members keep their views and reservations to themselves and agree with views they are opposed to. Members of the software development team and the project manager publicly agreed with each other, while privately disagreeing. Thus, the fourth

Table 2 Occurrences of the Abilene Paradox in telecoms team

Abilene Paradox symptom	Occurrence in case study
Members, as individuals, privately agree on the correct decision to make. This is not shared with the group.	Each developer agreed with the decision to adopt Agile, yet would not push for the decision to be implemented.
Members, as individuals, privately agree on how the problem or situation being addressed can be resolved. This is not shared with the group.	Each developer stated how Agile should have been adopted, but individually – not in the group.
Instead of communicating their views, members keep their views and reservations to themselves, agreeing with views they are opposed to.	Although initially and subsequently agreeing with the decision to adopt Agile, developers did not express opinions when Agile use gradually died.
As the individuals have not presented their views and reservations, a collective decision is made that is actually contrary to the views of all members.	The resistance to the decision occurred collectively, yet it went against the individual opinions.
Members feel frustration, even anger, at this and find someone, or some people, to blame.	The team blamed the team lead and the team lead blamed the developers.

symptom follows in that, as group members did not present their views and/or reservations, a collective decision is made that is actually contrary to the views of all members. The decision not to proceed with ASD in this project was arrived at collectively, against the individual wishes of team members. The fifth and final symptom is that group members feel frustration, even anger, at the situation they find themselves in and attribute blame to some external entity. In the case study, the software development project team as a whole placed the blame on the company's Quality Assurance Group. Subsequently, team members placed the blame on the project manager, while he reciprocated and blamed team members. In the final analysis, they were all responsible by virtue of not being open and honest with each other.

Conclusions and practical recommendations

We argue that the empirical findings presented above offer corroboration for this study's assumption that effective decision making by Agile teams is a function of the level of cohesion existing in a team and the subsequent level of empowerment in decision making bestowed upon the team by the project manager. The findings also support the assumption that the desire for continued team cohesion, coupled with a reluctance by team members to express views contrary to what are assumed to views of other team members and not just the project manager, leads to dysfunctional and ineffective decision making, such as groupthink and the Abilene Paradox. These assumptions lead to the following proposition for test by future researchers and consideration by practitioners:

Proposition 1 *Empowering a highly cohesive Agile team to make project-related decisions collectively leads to ineffective or dysfunctional decision making.*

A logical deduction from these assumptions, the above proposition, and theory on group psychology, leads to a second proposition viz.,

Proposition 2 *Empowering a newly formed or diverse Agile team to make project-related decisions collectively leads to effective and functional decision making.*

Of course, it would be impractical to suggest that new teams be established for every Agile project, as the many benefits of otherwise efficient, effective, and cohesive software teams would be lost. Similarly a team made up of diverse individuals may not bring the benefits associated with Agile teams; it may be contrary to the Agile need for cohesion. Hence, the following subsection offers practical recommendations based on an analysis of theory in organisational and group psychology.

Practical recommendations

It has long been recognised in IS research that conflict in information systems development projects can be beneficial to both process and product (Euchner *et al.*, 1993; Robey *et al.*, 1993). The literature on participative design also implicitly reflects this view (Ehn, 1988). It is with this in mind that we now consider possible solutions to the problems reported in this study on the adoption of ASD approaches.

Janis (1972) proposes several solutions to the problem of groupthink, based on the existence or stimulation of intra-group conflict. Two of Janis' recommendations are noteworthy, as they refer specifically to solutions to ineffective decision making viz.:

- Separate groups should be formed, under different leaders, to propose solutions to the same problem.
- A devil's advocate should be appointed: interestingly, Kanter (2001) also suggests this as the solution to the Abilene Paradox.

The question of whether these solutions would work in the adoption of an ASD remains to be answered. Extant research on ineffective decision making in reference disciplines supports the use of both of these approaches; however, this has not found its way into research on Agile software teams.

As this section proposes practical solutions to the problems posed by the study's findings, it would be a wasteful use of resources for an organisation to establish two Agile teams to solve the same problem, as per Janis' first recommendation. From a research perspective, researchers could set up a controlled experiment involving an experimental and a control group to investigate decision making. Alternatively, two or more cases could be purposefully selected to qualitatively test Propositions 1 and 2 (cf. Yin, 2003) or quantitative researchers could conduct survey-based research on a wider population of organisations.

Research aside, we contend that the use of a devil's advocate approach would be beneficial to practitioners and of interest to researchers going forward. Support for such an approach comes from studies such as Schweiger *et al.* (1989), and Herbert & Estes (1977). The primary goal of a devil's advocate is to examine and challenge any assumptions that decision makers hold (Schwenk, 1998). The reason for this is that, in cohesive groups, assumptions are made without reflection or discussion due to the tacit suppression of opinion, which contradicts the group's agreed perspective (Ottaviani & Sorensen, 2001). Researchers who argue for the use of devil's advocate in such situations assume that any decision that can withstand critique is a good decision. In the context of the present study, it was found that the Telecommunications Team decided collectively that the organisation's Quality Group was to blame for the failure to fully implement Agile. However, when questioned on the issue using feedback from the Quality Group, team members admitted that the fault lay elsewhere. Thus, critique or conflict reduces the likelihood of a faulty consensus in decision making (Cosier, 1981).

There are concerns that the use of conflict in the devil's advocate approach could create problems within cohesive teams (cf. Nemeth *et al.*, 2001; Nemeth and Goncalo, 2004). For example, a devil's advocate approach may not be advisable if it has a negative influence on any positive benefits that cohesion brings to an Agile team. However, Schweiger *et al.* (1989) found that the use of devil's advocate did not affect satisfaction in groups. In contrast, Nemeth *et al.* (2001) found that antipathy was created in such situations; nevertheless, studies have found that by properly controlling conflict, a healthy balance can be achieved – see, for example, Herbert & Estes (1977), who found that depersonalising conflict reduced antipathy and other negative consequences for a group. In addition, Sambamurthy & Poole (1992) discuss how conflict can be used beneficially in group scenarios; they (*ibid.* p. 225) argue that 'to take advantage of group strengths, conflict must be handled in a way that diverse perspectives are not stifled, members' commitment is maintained, and group cohesiveness is built'.

Thus, the introduction of 'conflict' though the devil's advocate approach can provide several benefits in Agile adoption decisions (cf. Euchner *et al.*, 1993 and Robey *et al.*, 1993, on conflict in traditional projects). Drawing on this conclusion, this article argues that a properly informed Agile project manager can act as a facilitator and fill the role of devil's advocate with beneficial project management outcomes. He or she could do this by questioning the assumptions underpinning the decisions made by his/her team and surface latent issues hidden due to groupthink, the Abilene Paradox, or other socio-psychological problems. In order to help practitioners in this regard, we present recommendations by MacDougall & Baum (1997) on the role of devil's advocate. MacDougall and Baum argue that the devil's advocate should:

- play the role consistently, that is they cannot switch from being the devil's advocate to contributing to the discussion;
- contribute in an assertive manner without dominating or forcing the discussion;
- ask the group how a different group would approach the same issue, or frame questions in a different way;
- alert the group to any leap from problem to solution that neglects argument and different perspectives.

We argue that by following the role of devil's advocate as presented here, the problems with the two Agile projects described above may have been avoided. For example, the recommended role of a project manager alerting members of his/her software team that any leap from problem to solution is inadvisable, would in all likelihood have benefited both teams, as the solution to the problems they were experiencing with Agile was to dilute their support for, and use of, the approach. Thus, what MacDougall & Baum (1997) describe as a leap from problem to solution, was a leap too far for these Agile teams.

Given the forgoing discussion, we now suggest a third and final proposition for test by future researchers and consideration by practitioners:

Proposition 3   *Empowering a highly cohesive Agile team to make project-related decisions collectively, but which are facilitated by a devil's advocate, leads to effective and functional decision making.*

The three propositions presented herein should, therefore, prove useful to researchers interested in testing (confirming or falsifying) the findings of this study or who are interested in investigating the phenomenon in other Agile teams.

Finally, this study does not claim that the decision to adopt an Agile approach for software development by

cohesive project teams will always result in dysfunctional decision making. Yin's (2003) technique of analytic generalisation, where the findings of the study were analysed with respect to extant theory in the IS and reference disciplines, does offer support for the conclusions presented above. Furthermore, although the research presented herein is based on ASD, there are ramifications for other areas of IS characterised by consensus in decision making, such as decisions made by cohesive teams in Theory Y-type organisations. In conclusion, the central argument of this article is that the Agile approach to project management can increase the likelihood of

problems with team-based decisions. We do not imply that Agile project management or Agile teams are inherently dysfunctional when it comes to decision making. There are, as indicated, many positive benefits of empowering and trusting a cohesive software development team – in fact, the benefits may outweigh any problems that arise. It is clear, however, that project management of Agile initiatives can be improved by accepting that groupthink and the Abilene Paradox can arise in teams, and that such problems can be avoided through the conscious application of a devil's advocate approach to decision making in Agile projects.

About the authors

John McAvoy is a lecturer in Business Information Systems, University College Cork, Ireland. John's research focuses on the management of small teams within Information Systems Development projects, specifically those using an Agile methodology. His research has been published in journals such as *Information and Software Technology*, *Journal of Information Systems Education*, *Journal of Decision Systems*, and in the proceedings of major international conferences such as ICIS and ECIS. Prior to lecturing, John had a variety of roles in the Information Systems field. He has worked across different industry sectors with roles ranging from systems administration in an aircraft design company to managing software development teams in telecommunications projects.

Tom Butler is a senior lecturer in Business Information Systems, University College Cork, Ireland. Tom's research focuses on investigating the origins of firm-level IT capabilities and the design, development, and implementation of information systems. He was lead researcher and project manager on two major action research-based initiatives on the design, development and deployment of IT-enabled knowledge management systems (KMS). Since 2005, Tom has been conducting research into the design, development, and implementation of Green IS. His research has been published in the *Information Systems Journal*, the *Journal of Strategic Information Systems*, the *Journal of Information Technology* and in the proceedings of major international conferences such as ICIS, ECIS, and IFIP 8.2 and 8.6.

References

ABRAHAMSSON P, SALO O, RONKAINEN J and WARSTA J (2002) *Agile software development methods* Review and analysis VTT Publications, Finland.

ARGYRIS C (1976) Single-loop and double-loop models in research on decision making. *Administrative Science Quarterly* **21(3)**, 363–375.

ARGYRIS C (1998) Empowerment: the emperor's new clothes. *Harvard Business Review* **76(3)**, 98–105.

AUER K, MEADE E and REEVES G (2003) The rules of the game. In *Extreme Programming and Agile Methods – Xp/Agile Universe 2003* (MAURER F and WELLS D, Eds), pp 35–42, Springer-Verlag, Berlin, Germany.

AUGUSTINE S (2005) *Managing Agile Projects*. Prentice Hall, USA.

BALTHAZARD P, POTTER R and WARREN J (2004) Expertise, extraversion and group interaction styles as performance indicators in virtual teams. *The Database for Advances in Information Systems* **35(1)**, 41–64.

BASADUR M (1997) Managing creativity: a Japanese model. In *The Human Side of Managing Technological Innovation* (KATZ R, Ed), pp 68–79, Oxford University Press, NY, USA.

BEAL D, COHEN R, BURGE M and MCLENDON C (2003) Cohesion and performance in groups: a meta-analytic clarification of construct relations. *Journal of Applied Psychology* **88(6)**, 989–1004.

BØDKER K and PEDERSEN J (1991) Workplace cultures: looking at artifacts, symbols and practices. In *Design at Work: Collaborative Design of Computer Systems* (GREENBAUM J and KING M, Eds), pp 121–136, Lawrence Erlbaum Associates, NJ, USA.

BOEHM B and TURNER R (2003) Rebalancing your organizations agility and discipline. In *Extreme Programming and Agile Methods – Xp/Agile Universe 2003* (MAURER F and WELLS D, Eds), pp 1–8, Springer-Verlag, Berlin, Germany.

BOEHM B and TURNER R (2004) *Balancing Agility and Discipline*. Pearson Education, MA, USA.

CARTWRIGHT D and ZANDER A (1968) Pressures to uniformity in groups. In *Group Dynamics* (CARTWRIGHT D and ZANDER A, Eds), pp 139–151, Harper & Row Publishers, NY, USA.

CARTWRIGHT R (2002) *Mastering Team Leadership*. Palgrave Macmillan, Wales.

CHIN G (2003) *Agile Project Management: How to Succeed in the Face of Changing Project Requirements*. AMACOM, NY.

COHN M (2004) *User Stories Applied for Agile Software Development*. Addison-Wesley, MA, USA.

COSIER R (1981) Dialectical inquiry is strategic planning: a case of premature acceptance? *The Academy of Management Review* **6(4)**, 643–648.

EHN P (1988) *Work-Oriented Design of Computer Artefacts*. Arbetslivscentrum, Stockholm, Sweden.

ERICKSON J, LYYTINEN K and SIAU K (2005) Agile modelling, Agile software development, and Extreme Programming: the state of research. *Journal of Database Management* **16(4)**, 88–100.

EZEY P (2003) Integration and its challenges in participant observation. *Qualitative Research* **3(2)**, 191–205.

EUCHNER J, SACHS P and The Nynex Panel (1993) The benefits of internal tension. *Communications of the ACM* **36(4)**, 53.

GOETZ J and LECOMPTE M (1984) *Ethnography and Qualitative Design in Educational Research*. Harcourt Brace Jovanovich, NY, USA.

GOLEMAN DP (1996) *Vital Lies Simple Truths: The Psychology of Self-Deception*. Simon and Schuster Books, NY, USA.

GOWDA R and CHAND D (1993) An exploration of the impact of individual and group factors on programmer productivity. *ACM conference on Computer Science* 338–345, ACM Press, Indianapolis, USA.

HARE A (1994) Individual versus group. In *Small Group Research: A Handbook* (HARE A, BLUMBERG H, DAVIES M and KENT M, Eds), pp 261–270, Ablex Publishing, NJ, USA.

HARVEY J (1974) The Abilene paradox: the management of agreement. *Organizational Dynamics* **3(1)**, 63–80.

HARVEY J (2001) Abilene revisited: An epilogue. *Organizational Dynamics* **17(2)**, 35–37.

HAZZAN O and TOMAYKO J (2003) The reflective practitioner perspective in Extreme Programming. In *Extreme Programming and Agile Methods – Xp/Agile Universe 2003* (MAURER F and WELLS D, Eds), pp 51–61, Springer-Verlag Berlin, Germany.

HERBERT T and ESTES R (1977) Improving executive decisions by formalizing dissent: the corporate devil's advocate. *The Academy of Management Review* **2(4)**, 662–667.

HIGHSMITH J (2004) *Agile Project Management*. Pearson Education, MA, USA.

HOGG M and HAINS S (1998) Friendship and group identification: a new look at the role of cohesiveness in groupthink. *European Journal of Social Psychology* **28(3)**, 323–341.

HUCZYNSKI A and BUCHANAN D (1991) *Organizational Behaviour. An Introductory Text* 2nd edn Prentice Hall, NY.

JANIS I (1972) *Victims of Groupthink*. Houghton Mifflin Company, MA, USA.

JORGENSEN D (1989) *Participant Observation: A Methodology for Human Studies*. Sage Publications, CA, USA.

KANTER R (2001) An Abilene defense: commentary one. *Organizational Dynamics* **17(2)**, 37–40.

KIM Y (2001) A comparative study of the 'Abilene paradox' and groupthink. *Public Administration Quarterly* **25(2)**, 168–187.

LANDY F and CONTE J (2004) *Work in the 21st Century*. McGraw-Hill, NY, USA.

LANGFRED C (2004) Too much of a good thing? Negative effects of high trust and individual autonomy in self managed teams. *Academy of Management Journal* **47(3)**, 385–399.

LEVINE J and MORELAND R (1990) Progress in small group research. *Annual Review Of Psychology* **41(1)**, 585–634.

MACDOUGALL C and BAUM F (1997) The devil's advocate: a strategy to avoid groupthink and stimulate discussion in focus groups. *Qualitative Health Research* **7(4)**, 532–541.

MANZ C and NECK N (1995) Teamthink: beyond the groupthink syndrome in self-managing work teams. *Journal of Managerial Psychology* **10(1)**, 7–15.

MANZ C and SIMS H (1987) Leading workers to lead themselves: the external leadership of self-managing work teams. *Administrative Science Quarterly* **32(1)**, 106–128.

MCAVOY J and SAMMON D (2006) Towards the development of a simple tool to assist in agile methodology adoption decisions: Agile adoption matrix. In *The Proceedings of the European Conference of Information Systems 2006* (LJUNGBERG J and ANDERSSON M, Eds) pp 2101–2110, Gothenburg, Sweden.

MCBREEN P (2002) *Questioning Extreme Programming*. Pearson Education, NJ, USA.

MCGRATH J (1984) *Groups, Interaction, and Performance*. Prentice-Hall, NJ, USA.

MCGREGOR D (1960) Theory X and theory Y. In *Organization Theory* (PUGH D, Ed), 3rd edn, pp 358–374, Penguin Books, London, UK.

MINTZBERG H, RAISINGHANI D and THÉORÊT A (1976) The structure of 'unstructured' decision processes. *Administrative Science Quarterly* **21(2)**, 246–275.

NECK C (1996) Letterman or leno: a groupthink analysis of successive decisions made by the national broadcasting company. *Journal of Managerial Psychology* **11(8)**, 3–17.

NEMETH C and GONCALO J (2004) Influence and persuasion in small groups. In *Persuasion: Psychological Insights and Perspectives* (SHAVITT S and BROCK TC, Eds), pp 171–194, Allyn and Bacon, MA, USA.

NEMETH C, BROWN K and ROGERS J (2001) Devil's advocacy versus authentic dissent: stimulating quantity and quality. *European Journal of Social Psychology* **31(6)**, 707–720.

OTTAVIANI M and SORENSEN P (2001) Information aggregation in debate: who should speak first. *Journal of Public Economics* **81(3)**, 393–421.

PATCHING K (1999) *Management and Organisational Development*. Macmillian Business, London, UK.

PATTON M (1990) *Qualitative Evaluation and Research Methods*. Sage Publications Ltd, London, UK.

ROBBINS H and FINLEY M (1998) *Why Teams Don't Work*. Orion Publishing, London, England.

ROBEY D, SMITH LA and VIJAYASARATHY LR (1993) Perceptions of conflict and success in information systems development projects. *Journal of Management Information Systems* **10(1)**, 123–139.

ROBINSON H and SHARP H (2005) The social side of technical practices. *Proceedings of 6th Processes in Software Engineering 5th International Conference* (BAUMEISTER H, ECKSTEIN J and HOLCOMBE M, Eds), pp 100–108, Springer-Verlag, Berlin, Germany.

ROOCK S and WOLF H (2004) Agile project controlling. In *Extreme Programming and Agile Processes in Software Engineering 5th International Conference* (ECKSTEIN J, BAUMEISTER H and HANSEN R, Eds), pp 202–209, Springer-Verlag, Berlin, Germany.

SALO O, KOLEHMAINEN K, KYLLONEN P, LOTHMAN J, SALMIJARVI S and ABRAHAMSSON P (2004) Self-adaptability of agile software processes: a case study on post-iteration workshops. In *Extreme Programming and Agile Processes in Software Engineering 5th International Conference* (ECKSTEIN J, BAUMEISTER H and HANSEN R, Eds), pp 184–193, Springer-Verlag, Berlin, Germany.

SAMBAMURTHY V and POOLE MS (1992) The effects of variations in capabilities of GDSS designs on management of cognitive conflict groups. *Information Systems Research* **3(3)**, 224–251.

SCHUH P (2004) *Integrating Agile Development in the Real World*. Delmar Thomson Learning, NY, USA.

SCHWEIGER D, SANDBERG W and RECHNER P (1989) Experiential effects of dialectical inquiry, devil's advocacy, and consensus approaches to strategic decision making. *Academy of Management Journal* **32(4)**, 745–772.

SCHWENK C (1998) Effects of devil's advocacy on escalating commitment. *Human Relations* **41(10)**, 769–782.

SPRADLEY J (1980) *Participant Observation*. Holt, Rinehard, and Winston, NY, USA.

STATT D (2004) *Psychology and the World of Work*. Palgrave Macmillan, Hampshire, UK.

STEPHENS M and ROSENBERG D (2003) *Extreme Programming Refactored: The Case Against XP*. Springer-Verlag, Heidleberg, Germany.

THAMHAIN H and WILEMON D (1997) Building high performance engineering project teams. In *The Human Side of Managing Technological Innovation* (KATZ R, Ed), pp 125–136, Oxford University Press, NY, USA.

THOMAS F (1990) Leadership, the development cycle, maturity & power. *Proceedings of the 18th Annual ACM SIGUCCS Conference on User Services* (SARRI S, Ed), pp 399–402, ACM Press, NY, USA.

VON BERGEN C and KIRK R (1978) Groupthink: when too many heads spoil the decision. *Management Review* **67(3)**, 44–49.

YIN RK (2003) *Case Study Research: Design and Method*. Sage Publications, Thousand Oaks, CA.