The Effectiveness of Software Development Technical Reviews: A Behaviorally Motivated Program of Research

Chris Sauer, D. Ross Jeffery, *Member*, *IEEE*, Lesley Land, and Philip Yetton

Abstract—Software engineers use a number of different types of software development technical review (SDTR) for the purpose of detecting defects in software products. This paper applies the behavioral theory of group performance to explain the outcomes of software reviews. A program of empirical research is developed, including propositions to both explain review performance and identify ways of improving review performance based on the specific strengths of individuals and groups. Its contributions are to clarify our understanding of what drives defect detection performance in SDTRs and to set an agenda for future research. In identifying individuals' task expertise as the primary driver of review performance, the research program suggests specific points of leverage for substantially improving review performance. It points to the importance of understanding software reading expertise and implies the need for a reconsideration of existing approaches to managing reviews.

Index Terms—Inspections, walkthroughs, technical reviews, defects, defect detection, groups, group process, group size, expertise, reading, training, behavioral research, theory, research program.

1 Introduction

Software quality is an important issue for software engineers, business managers, and researchers [18], [51], [83], [84], [85], [86], [87]. Software reviews, such as inspections, walkthroughs, and technical reviews, are one widely practiced approach to improving quality. The central insight common to the different types of review, which we refer to generically as software development technical reviews (SDTRs), is that a group of software engineers, meeting together to review a software product, improves its quality by detecting defects which otherwise would have gone unnoticed by the product's author.

In practice, except where zero defects are required for safety critical applications, software engineers and managers have to trade the number of defects detected against cycle time and resource costs. Cycle time is increased by preparation time and by delays scheduling review meetings [8], [72]. Resource costs for a review increase as a function of the number of reviewers. In this paper, we are principally concerned with understanding the effectiveness of SDTRs in detecting defects. Once this is understood, it becomes easier to make informed trade-offs between effectiveness, cycle time, and resource costs.

- *C. Sauer is with Templeton College, Oxford University, Oxford, OX1 5NY, U.K. E-mail: Chris.Sauer@templeton.oxford.ac.uk.*
- *P. Yetton is with the Fujitsu Centre for Managing Information Technology in Organisations, Australian Graduate School of Management, The University of New South Wales, Sydney, NSW 2052, Australia. E-mail: P.Yetton@unsw.edu.au.*
- *D.R. Jeffery and L. Land are with the Centre for Advanced Empirical Software Research, School of Information Systems, The University of New South Wales, Syndney, NSW 2052, Australia. E-mail: {R.Jeffery, L.Land}@unsw.edu.au.*

Manuscript received 25 Oct. 1996; revised 3 June 1998; accepted 13 Nov. 1998.
Recommended for acceptance by H. Muller.
For information on obtaining reprints of this article, please send e-mail to: tse@computer.org, and reference IEEECS Log Number 101257.

1.1 Focus

We take it as an assumption that SDTRs are effective in the most basic respect of finding more defects than no review at all. This is the collective understanding and belief of the practitioner and research communities [13], [17], [78], [88]. How much more effective they are than this most basic requirement is an empirical question. Our concern is to ask *what makes them as effective as they are* because we want to establish a systematic basis for discovering ways to improve defect detection performance. Without this, it is difficult to make sound decisions about the relative potential of such diverse interventions as N-fold inspection [47], [60], phased inspections [31], checklists and scenarios [33], [56], [57], and computer support [45].

1.2 Motivation for a Research Program

We develop a *research program* because of the benefits it offers by providing a structure for the way we develop knowledge. A scientific research program is driven by a core theoretical insight supported by subsidiary theoretical statements. For example, the program of cognitive science has been driven by the theoretical insight that the human brain/mind functions in ways analogous to a digital computer's hardware/software. A typical subsidiary theoretical statement would be that human understanding of natural language requires domain specific data structures in the form of scripts. A scientific research program has three strengths. First, the underlying theory stimulates new ways of thinking about a problem. Second, it offers new explanations and predictions. Third, when predictions are not confirmed, the underlying theory provides a source of alternative explanations for exploration [32]. Thus, when

TABLE 1
Common and Variant Features of SDTRs (Adapted from Kim et al. [30])

	Common Features	Variations
Objective	defect detection	error correction, cost effectiveness
	secondary benefits	predictable software, productivity, establishment of standards, project control, education, training, improved competence, user satisfaction, professionalism
Product	any software product in the life cycle	process aids including tools, methods, standards
Process	two stage:	purpose of each stage, number of meetings, number of teams, aids, team formation, training
	1) individual	
	2) group	
Group Meeting	roles, specifically moderator	types of role, role definition
	interaction	meeting duration, team size, management involvement, use for evaluation
Group Outputs	reports	report type, contents, format, use

scripts proved too limited a basis for a general explanation of natural language understanding, the program of cognitive science provided ready alternatives in the form of more sophisticated programs and data structures.

Our research program is based on a theory of group performance and consists of a series of testable propositions. These are structured to answer three important questions. First, because the underlying theory is so important in directing further research, we seek to validate it by asking:

1. What makes SDTRs effective at defect detection?

This allows us to determine how and to what degree the behavioral theory is applicable to SDTRs. Once the theory is validated in this context, ways of improving the performance of the conventional two-stage, individual-group review become apparent. To do this, we present propositions which address the question:

2. How can we improve SDTR performance within the current review design?

We then take advantage of the program's capacity to generate novel ideas by developing propositions which address the question:

3. What design alternatives would theory predict to be more effective?

1.3 Contributions

In doing the above, the paper makes two contributions. *First*, it clarifies researchers' and practicing software engineers' understanding of what drives defect detection performance in SDTRs. By emphasizing the importance of individuals' expertise in the performance of a review, it explains why reviews achieve the level of performance that they do and suggests how they can be improved. *Second*, the paper sets an agenda for SDTR research with the objectives of validating the underlying theory, confirming the principal sources of improved performance, and developing design alternatives.

1.4 Structure of Paper

Section 2 reviews empirical research to date on SDTRs. It recognizes variability in SDTR designs and shows that empirical research on reviews has not yet developed a general explanation of performance. It identifies the need for cumulative research findings which can underwrite improved review performance. Section 3 describes the structure of the behavioral theory of group performance, its findings, and its applicability to research into SDTRs. Section 4 uses the behavioral theory to develop our program in the form of a series of propositions. Section 5 discusses the propositions in the context of current empirical research and draws on this to develop implications for research and practice. Section 6 draws conclusions.

2 Current Research on SDTRs

There are various designs for SDTRs. In a survey of the seminal literature on inspections [13], [14], walkthroughs [78],

and technical reviews [17], [74], Kim et al. [30] found a limited set of common features (Table 1).

The common features form the basis for a wide ranging characterization of SDTRs as:

> an organizational device for *detecting defects* in software *products* at any stage of the life cycle and for obtaining *secondary benefits* through a *two-stage process* in which software engineers first *independently inspect* the software product for defects and then *combine their efforts* in a *group meeting* in which the participants adopt *roles* with the goal of producing a *report* in which all the defects agreed upon by the group are identified.

This characterization serves as the basis for the model of the SDTR task (the task model) on which our research program focuses. Stage one, individual preparation, is a multi-item discovery task, which means that reviewers must seek out multiple defects where the number of defects is unknown. Stage two, the meeting stage, involves reviewers interacting in a group to settle on an agreed set of defects. Defects identified in review are often called *issues*. Issues include *false positives* (i.e., erroneously identified defects). While review performance at individual and group stages can be measured in terms of true defects detected, false positives should also be included. In addition, there are matters of style and standards which, for simplicity, we do not consider further [55].

Our task model represents a suitable baseline for research. It is similar to that employed in several recent studies [11], [33], [34], [55]. Its focus is defect detection because most organizations would not conduct reviews solely for the secondary benefits, such as education of developers. It does not extend to defect correction, or rework, because that is beyond the scope of the behavioral theory on which we base our proposals.

Research on SDTRs has been wide-ranging, covering organizational adoption of reviews [23], training [1], [15], [66], economic models [16], review designs [13], [20], [78], and automation [45]. Much of this literature is constructive [58] in that it develops, tests, and evaluates diverse new interventions and designs, such as N-fold inspection [47], [60] and phased inspections [31], or new methods and tools, such as checklists, scenarios, and computer support [29], [33], [56], [57]. The determinants of review performance have been addressed in a number of different ways [34], [55], [75], but empirically based conclusions have usually been relative to cost and cycle time. The determinants of defect detection remain unclear. For example, 20 years after Fagan's seminal paper [13], debate continues over whether a key component of inspections, the group meeting, is necessary for defect detection because it is not clear whether, in the context of a process in which individual preparation focuses on defect detection, significant numbers of defects are discovered through reviewers interacting (synergy) [11], [13] ,[33], [34], [55], [72].

The lack of an underlying theory of review performance has meant that we have not had a systematic basis for deciding research priorities among potential determinants of performance. For example, in advance of exploratory research, any decision as to which of, say, computer support for inspections or software reading is likely to lead to more significant performance gains would be arbitrary. Lack of theory has also made it difficult to choose among competing explanations when predictions have not been supported. For example, Mashayekhi et al. [48] appeared surprised that, in their study, the relative performance of review teams was not affected by computer support. The behavioral theory that we adopt for our program provides a simple explanation, viz that individual expertise drives group performance and the experimental manipulation did not vary the expertise of the teams. Overall, lack of theory has meant that research has been cumulative only on highly specific topics where groups of researchers have conducted a number of studies on a single topic, see, for example, the work of Porter, Votta and others on the value of task aids and the elusiveness of group synergy [33], [56], [57]. However, existing research can be reframed to contribute to a structured and cumulative development of knowledge, as we demonstrate in Section 5.

3 The Behavioral Theory of Group Performance

Researchers in the behavioral sciences have studied group performance for more than 50 years—work by Shaw and Thorndike dates back to the 1930s [61], [67]. Studies have ranged over a variety of facets of group behavior. However, it was only in the 1980s that a broad range of prior research was unified within a concise, explanatory theory [6], [7], [76], [77] whose core ideas have been further developed in the 1990s [19]. We refer to this theory as *the behavioral theory of group performance*.

The behavioral theory has been developed through laboratory studies on a two-staged task which includes an individual preparation stage and a stage during which the group meets. The problem studied has been a multi-item judgement task where subjects know how many items must be judged. An example of this kind of task is the "Lost in the Desert" problem in which subjects are required to imagine themselves stranded in the desert with a limited number of implements available to them, e.g., knife, string, mirror, etc., and they are required to rank them in order of their survival value. Individuals make their own judgements and then meet to decide a group ranking. The theory is powerful in the sense that it explains a high degree of group performance variance [76], [77].

The behavioral theory's origins in the laboratory may appear to limit its applicability to SDTR research. However, the fact that the theory has been developed in the laboratory should not detract from its external validity since laboratory experiments on behavior generalize to the field better than is usually supposed [39]. Moreover, the theory has been successfully applied to audit reviews which are similar to the defect detection task [68], [69], [70], [71]. Audit reviews involve defect discovery where it is not known how many defects there are in the audit document. And, as with reviews of software requirements, there may not always be an objectively correct solution. Table 2 summarizes the similarities between the SDTR and behavioral task models. These similarities suggest there are good grounds for adopting the behavioral theory as the underlying theory for a research program on SDTR performance.

TABLE 2
Comparison of SDTR and Behavioral Task Models

SDTR Task Model	Behavioural Task Model
two stage task	two stage task
multi-item task - number of defects unknown	multi-item task - number of sub-problems known. Successfully applied to audit task where number of sub-tasks is unknown
first stage individual preparation - discovery of defects including false positives	first stage individual preparation - individual judgement for each sub-problem
second stage interacting group - group discrimination judgement for each proposed defect	second stage interacting group - group judgement for each sub-problem

3.1 Structure of the Behavioral Theory

Fig. 1 provides a schematic summary of the behavioral theory as it has been developed with the task model described. The most salient finding of the empirical research on which the theory is based is that *group performance is dominated by the available task expertise* [77]. The performance of an interacting group can be closely modeled by: 1) taking the performance of its individual members and 2) pooling their performance according to a nonequal weighted decision scheme favoring high task expertise, i.e., by combining individuals' performance on a problem in a way that gives greater weight to the performance of the most expert individual in the group. Since both the performance of the individuals and the weighting scheme relate only to task expertise, the group's performance is seen to be strongly determined by expertise. The strategy the group adopts for using the expertise available to it, i.e., how it recognizes expert from nonexpert solutions, is its *social decision scheme*. An important related finding is that interacting groups do not achieve synergies. They do not generate a significant volume of new, creative problem solutions (sometimes called emergents) beyond those already generated in the individual phase of the task [7], [76].

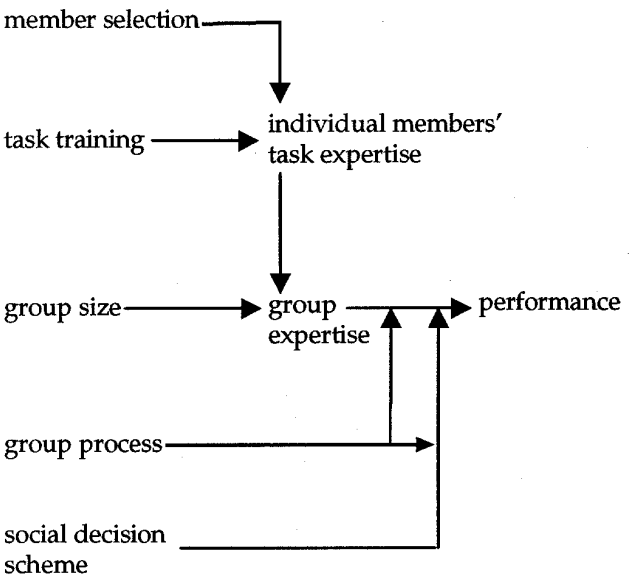

Fig. 1. Schematic summary of the behavioral theory of group performance.

Group expertise is constituted from the task expertise of individual members and is a function of *group size*—additional group members add to available expertise [77]. It follows that, for a group of given size, appropriate *selection* of group members can increase the level of group expertise. In addition, *task training* can improve individuals' expertise and, hence, improve the group's performance [6], [19], [27].

How the group members' expertise contributes to its performance on a single item of a multi-item problem is explained by the social decision scheme used. Decision schemes determine how the group applies available expertise and are operationalized in terms of *plurality effects*. These occur when multiple group members agree on the solution to a problem. Behavioral research has found that, where a majority agrees on a particular solution, that solution is adopted by the group [7]. In the absence of a majority, a correct or accurate solution agreed on by just two members is typically sufficient.

If there is no agreement, then the quality of *group process* in resolving the conflicting opinions determines the social decision scheme adopted by the group. Group process refers to the processes of interaction among the group members. These include the social processes through which information is shared and used, including communication, turn-taking in discussion, and conflict resolution. The better the group process, the more likely members are to be able to share relevant information and weigh it so that a better solution is identified and accepted [7], [21]. Poor process makes this difficult and, so, the group is more likely to operate as if all members' views were of equal weight (equal weight decision scheme)—they simply average out members' preferred solutions with the result that they perform worse than if they had relied solely on their best member.

The behavioral theory has the deceptive appearance of being just common sense. Its distinction is that it is explicit about how a number of different facets of groups, such as size, composition, and process, affect performance through

their influence on the application of expertise. The theory identifies some common misconceptions about groups such as that they necessarily result in more creative problem-solving and that group process is the most important determinant of performance.

3.2 Behavioral Research Findings

Behavioral research supports the widely held belief that groups perform better than individuals. The reason is that groups have more expertise available. As a new member is added, a group's expertise increases and, therefore, on average, its performance rises. This is subject to decreasing returns to scale as the expected increment in performance obtained from adding each new member to a group declines. Ultimately, group performance itself starts to decline as a result of *process loss* [65], [77], which is the difference between the group's predicted performance based on its available expertise and the group's actual performance. Its cause is put down to poor interactions among group members. The critical size at which loss begins to occur varies according to the task [65]. Not surprisingly, therefore, there is some variety in findings in regard to this limit [79], [65], [28], [35], [37]. For a problem such as Lost in the Desert, process loss has been identified for group sizes above four [77]. Where group members have high task expertise, performance benefits beyond two are insignificant [7] and, under circumstances where the best member of a group can be identified with confidence, using that individual's work alone may be an effective alternative to a group [38], [76]. Some recent studies suggest that this particular finding may not extend to *established* groups, i.e., groups whose members remain the same through many group tasks over an extended period of time [49], [73]. However, we doubt that this is especially important for software reviews because we have found established groups to be relatively rare, i.e., in one company, on only 15 occasions out of a sample of 118 reviews conducted over several years were a pair of reviewers working together for more than the fourth time (authors' unpublished field study). Our own experience suggests that high staff turnover often makes it difficult to develop established groups even where there is a policy to do so.

The behavioral theory's implications for SDTRs are that interventions which significantly increase the available defect detection expertise should have the largest impact on performance. Process interventions can be expected to be helpful only where there is available expertise but disagreement about its application. In this case, if process is poor, expertise may be lost. But, when expertise is poor, excellent process does not increase the available expertise and, hence, does not improve performance.

4 A Program of Empirical Research into SDTRs

The program is divided into three parts. The first is concerned with identifying the determinants of defect detection performance for SDTRs conducted in accordance with our task model. The second is concerned with understanding the potential and limits of two theoretically motivated sources of improvement given our task model. The third is concerned with exploring the potential for performance improvement under an amended task model.

4.1 Determinants of SDTR Performance

The purpose of Propositions 1 to 4 is to use the behavioral theory to make predictions as to what are and are not the key determinants of defect detection in SDTRs.

4.1.1 Expertise

Task expertise has been used in only a few cases to explain SDTR performance [63], [75]. In some cases, it appears to have been overlooked [48]. In other cases, experimentalists manipulating other variables control for expertise, thereby reducing their chance of identifying its role, see, for example, Lanubile and Visaggio [33]. If researchers manipulate the expertise variable, there is good reason to expect that it will prove the dominant determinant of performance. Formally, we propose:

P1 In SDTRs, task expertise is the dominant determinant of group performance.

4.1.2 Decision Schemes

Given the lack of attention to task expertise, it is not surprising that research on SDTRs has also not explored the decision schemes by which a group member's expertise is identified by the group. The behavioral theory predicts that, for SDTRs, a majority or plurality having discovered any issue is sufficient for the meeting to accept it as a true defect. Formally:

P2 In SDTRs, decision schemes (plurality effects) influence interacting group performance.

4.1.3 Group Process

When a defect has been identified by only one reviewer, it is unlikely that other reviewers already have a firm view on it. The behavioral theory predicts that, where there is no plurality by which to decide whether an issue is a true defect, a group's ability to make a correct discrimination is positively influenced by the quality of its group processes.

P3 In SDTRs, in the absence of a plurality, interacting group performance is a positive function of process skills.

4.1.4 Synergy

It is widely believed that SDTR meetings result in new and better outcomes than can be achieved by individuals. Votta [72] reports that 79 percent of a sample of 29 authors and managers thought synergy a reason why reviews were conducted. Behavioral studies have found no evidence of synergy as a source of group advantage [7], [76]. This implies, for SDTRs conducted in accordance with our task model, that the group meeting does not discover significant numbers of new defects beyond the aggregation of those discovered by individuals (often referred to as the nominal group):

P4 In SDTRs, the interacting group meeting does not improve group performance over the nominal group by discovering new defects.

4.1.5 Summary

The role of Propositions 1 to 3 is to confirm that, as the behavioral theory suggests, SDTR performance owes more to the expertise of the individual reviewers than any special quality of interaction in the review meeting. Proposition 4's role is to confirm that synergy does not constitute an independent justification of review meetings.

4.2 Improving SDTR Performance

It follows from the behavioral theory that SDTR performance would be influenced by interventions which vary the expertise available to the review group, specifically *selection*, *training*, and *groupsize*. The importance of reviewer selection for effectiveness follows from Proposition 1. The purpose of Propositions 5 to 7 is to validate the behavioral theory's predictions in relation to training and group size.

4.2.1 Task Training

Behavioral theory has found significant benefits from providing group members with generic problem-solving training. For SDTRs, training is widely recommended, but usually takes the form of process training to induct reviewers into the inspection process. Only occasionally is task training recommended to develop reviewers' defect detection skills [15], [59], [66]. Formally, we propose:

P5 In SDTRs, group performance is a positive function of task training

4.2.2 Performance/Size Relationship

Behavioral theory has found that group performance improves with increases in task expertise and, hence, improves as group size increases. In view of the effects of the number of reviewers on cost and cycle time, it would be valuable to identify the function linking defect detection performance and review group size. Behavioral theory has found that the relationship is a function of expertise. This may be complicated for SDTRs in respect to the need for different expertises required by some review designs [53], [78], but it is only a complication and does not alter the fundamental proposition:

P6 In SDTRs, the performance/size relationship is a function of task expertise.

4.2.3 Limit to Group Size

There is no agreement on the optimal size for SDTR groups. The normative literature varies in its recommendations [30]. For example, Madachy et al. [46] recommend five reviewers, Gilb [20] proposes three to five but sometimes more, while Grady [22] finds four to five reviewers optimal. Industrial practice varies even within a single enterprise, for example, sizes between four and 12 have been used at AT&T [11]. Because size is typically informed by cost-effectiveness considerations, it is not apparent whether size recommendations reflect any recognition that there will be a *limit* beyond which effectiveness will decline as a result of increased process loss outweighing the gains from increased expertise. Formally, we propose:

P7 In SDTRs, above a critical limit, performance declines with group size.

4.2.4 Summary

The role of Propositions 5 to 7 is to confirm that, as the behavioral theory suggests, the performance of reviews which conform to our task model can be enhanced through task training and appropriate group size. Propositions 6 and 7, if confirmed, will demonstrate that the relationship between group size and performance is not a simple linear function.

4.3 Design Alternatives

The purpose of Propositions 8 to 11 is to identify assumptions of the SDTR task model which might be relaxed and to explore some predicted performance effects of the implied design alternatives. Specifically, Proposition 8 seeks to identify the source of performance advantage of review meetings. Proposition 9 tests a minimalist alternative to meetings. Propositions 10 and 11 address the alternative when the performance advantage of meetings is negligible.

4.3.1 Advantage of Interacting over Nominal Groups

The SDTR task model takes for granted that review meetings improve defect detection performance. In the absence of synergies, and leaving aside task models in which defect discovery by individuals is discouraged [13], it follows from behavioral theory that this improvement will lie in the application of joint expertise to discriminate true defects from false positives [29].

It is an empirical question how significant a problem false positives are. There are mixed data. Most researchers are silent about false positives or merely remark that they have discarded them [33]. In a field study at AT&T, false positives proved to be insignificant [72]. By contrast, in a laboratory setting, they have been found to be significant [34], [81]. This difference may be plausibly explained by the different level of expertise of the subjects. The AT&T subjects were professional software engineers whose expertise we speculate would be among the highest in the industry. The laboratory study subjects were third year undergraduates. However, it is not possible to tell whether a similar explanation can be applied to Porter et al.'s [55] finding in an industrial study that 22 percent of defects reported by the group review were discarded by the software author as false positives. If level of expertise is the determinant of the occurrence of false positives, then it is an open question how widely occurring they are across different levels of expertise in industry. We predict:

P8 In SDTRs, the performance advantage of an interacting group over a nominal group is a function of the level of false positives discovered by individuals.

4.3.2 Expert Pairs

Behavioral research has found that expert pairs can be expected to perform as well as larger sized groups in judgment tasks [7]. In effect, if there is a high enough probability that reviewers will make accurate judgements, one reviewer acts as a check on the other, but no value is gained from having further reviewers. This principle is put into practice by auditors, where an audit manager reviews an audit senior's work and, in turn, this audit manager's

work is reviewed by an audit partner. Essentially, this decision model is an expert pair. By implication:

P9 In SDTRs, an expert pair performs the discrimination task as well as any larger group.

4.3.3 Defect Discovery

While meetings can add value, as noted in Section 3.2, they are also subject to process loss. Thus, while a review meeting may improve defect discrimination, it may cause some loss of performance by overlooking defects discovered by individual reviewers. It follows that nominal group designs outperform others in terms of the number of defects discovered. Formally, the proposition is:

P10 In SDTRs, nominal groups outperform alternatives at the discovery task.

4.3.4 Performance/Size Relationship for Nominal Groups

While an expert pair may be sufficient to discriminate true defects and false positives, it is doubtful whether, in general, individual reviewers are sufficiently expert at discovering defects that a pair would suffice for the discovery task. Several studies where the number of defects is known by the experimenters show that individual subjects, both students and professionals, discover in the order of only one in three defects [33], [34], [50]. Schneider et al. [60] find a similar level for three person teams. Indirect evidence from work on N-fold inspections [47], [60] suggests that increasing the number of reviewers does not quickly exhaust the defects remaining to be discovered.

As with the performance of the review meeting (Proposition 6), nominal group performance can be expected to be a function of size and individual expertise with decreasing returns to scale. Unlike the review meeting, because process loss is eliminated, the limit on discovery performance for a nominal group is defined not by group size but by the absolute number of defects in the software product. The detail of the relationship between the number of reviewers, their expertise, and their performance at defect discovery currently awaits empirical exploration.

P11 In SDTRs, the defect discovery performance/size relationship for nominal groups is a function of task expertise.

4.3.5 Summary

The role of Proposition 8 is to confirm that review meetings add value principally by discriminating true defects and false positives. Proposition 9 is to confirm that a two person review meeting is sufficient for the discrimination task if the reviewers are expert. Propositions 10 and 11 serve to confirm that discovery is most effective if done independently of a review meeting.

5 Discussion

5.1 Plausibility

The purpose of our program is to lay down a structure for future research. Full empirical testing will follow. However, there is already evidence in the software engineering literature, including our own empirical work, that confirms the initial plausibility of the program. Table 3 summarizes existing research as it relates to our program.

There is already some empirical evidence to support Proposition 1's contention that task expertise is the dominant determinant of SDTR performance. Siy [63] found in an industrial setting that one of the best predictors of group review performance was the presence of one of two high performing reviewers. In our own research, we have found that most of the true defects reported by a review group were discovered by an individual in preparation, rather than during the process of group interaction [34], [82]. This is reinforced by the considerable evidence accumulating in support of Proposition 4, that the performance advantage of interacting groups over individuals does not derive from the group discovering new defects in its meeting [11], [29], [33], [57], [72].

There is no empirical evidence currently available to support Proposition 2. Research has not typically examined cases where two reviewers detect the same defect in preparation. While the proposition is prima facie plausible, it is unclear from available data how often more than one reviewer finds the same defect [55]. Work on N-fold inspections suggests that i the order of 5 to 15 percent of all defects are found by more than one reviewer [47], [60]. Another study reported that, in a design review using eight reviewers, five out of 24 known defects were discovered by more than one reviewer [11]. However, our own field study of 110 three person reviews in an industrial setting found that less than 1 percent of all defects discovered were found by more than one reviewer (unpublished study). How representative any of these findings are remains to be determined.

Where the frequency of plurality is high in a review, it implies a high degree of shared expertise among the reviewers. Infrequent plurality could imply low expertise, or different and nonoverlapping expertises among the reviewers. It could also imply too large a product or too little time for the review. This suggests that we should explore what other decision schemes groups might use to determine relevant expertise. For example, where reviewers are recognized to have different skills, do group members defer to the recognized expert?

There is as yet no evidence directly relating to Proposition 3. That is, there are no results to demonstrate the influence of group process in the absence of plurality. A recent comparison of reviews with and without process roles (moderator, scribe, reader), and where plurality was rare, shows that roles improve process quality and group review performance [81].

As noted earlier in this section, the results from a number of recent studies support Proposition 4 by casting doubt on whether meetings discover significant numbers of new defects [11], [29], [33], [34], [57]. Fagan [13] reports apparently contrary data, but this is not surprising because, in his work, the interaction phase is dedicated to substantive defect detection, whereas the preparation task is confined to familiarization with the software product. Porter et al. [55] also have results suggesting significant emergent defects from review meetings, but

TABLE 3
Summary of Propositions and Related Literature

Proposition		Evidence in the Literature
P1	Task expertise is the dominant determinant of group performance	Weller [75] explains field performance in terms of expertise. Siy [63] finds the presence of 2 specific reviewers a strong predictor of group performance. Otherwise, ignored or deliberately factored out.
P2	Decision schemes (plurality effects) influence interacting group performance	Limited evidence about pluralities (overlap among defects detected by individuals) [11, 47, 55, 60, 81, authors unpublished field data].
P3	In the absence of a plurality, interacting group performance is a positive function of process skills	No relevant studies.
P4	The interacting group meeting does not improve group performance over the nominal group by discovering new defects	Eick et al [11], Votta [72], Porter et al [57], Johnson & Tjahjono [29], Lanubile & Visaggio [33], Land et al [34] find supporting evidence. Fagan [13], Porter et al [55] present variant findings.
P5	Group performance is a positive function of task training	Fowler [15], Redmill [59], Strauss & Ebenau [66] recommend training. Rifkin & Deimel [80] reduced customer-discovered defects by 90% through training in software reading.
P6	The performance/size relationship is a function of task expertise	Weller [75] finds performance an increasing function of size in the range 3 to 4. Porter et al [55] find 2 outperforms 1 but 4 does not outperform 2. Other data on performance/size relationship not rigorously reported.
P7	Above a critical limit, performance declines with group size	Porter et al [55], Lanubile & Visaggio [33], Lau et al [34], Land et al [82] find evidence of process loss. Madachy et al [46], Gilb [20], Grady [22] make size recommendations - usually in relation to cost-effectiveness.
P8	The performance advantage of an interacting group over a nominal group is a function of the level of false positives discovered by individuals	Proposition untested. Johnson & Tjahjono [29] find interacting groups have fewer false positives than nominals. Lau et al [34, 81] find interacting group has greater probability of eliminating false positives than true defects.
P9	An expert pair performs the discrimination task as well as any larger group	Not tested. Porter et al [55] find pairs perform as well as 4.
P10	Nominal groups outperform alternatives at the discovery task	Myers [50], Lanubile & Visaggio [33], Lau et al [34], Land et al [82] find detection rates which favour nominal groups for discovery. Martin & Tsai [47], Schneider et al [60] find levels of overlap favouring nominal groups for discovery. Lau et al [34] find nominal groups outperform their best member. Johnson & Tjahjono [29] find interacting groups do not outperform nominals at discovery.
P11	The defect discovery performance/size relationship for nominal groups is a function of task expertise	Martin & Tsai [47], Schneider et al [60] find performance increases with scale when defects are aggregated for N-fold groups. Relation to task expertise unstudied.

this may be partly explained in terms of the performance measures used.

The point about Proposition 5, that task training improves group performance, is not simply, as the literature already suggests, that training helps. Rather, it is that behavioral theory predicts that interventions that improve individual reviewers' ability to find defects and distinguish true defects from false positives improve

group performance. Software engineering research has not, for the most part, examined such targeted training. The only relevant study is highly supportive. It showed a 90 percent reduction in the defects reported by customers after software release through training reviewers in software reading techniques [80].

Although review group size has an important influence on cost and cycle time, the results relating to Proposition 6 do not present a clear picture. No studies have examined how the relationship between size and performance varies with expertise. We are aware of two studies which directly compare the performance of different sized groups. Using field data from three years' inspections, Weller [75] found four person inspection teams twice as effective at discovering defects as three person teams. This is consistent with the behavioral theory in showing review performance to be an increasing function of size within the range of group size three to four [77]. It is inconsistent with the theory to the extent that a 100 percent performance improvement does not reflect declining returns to scale. Given that the data are not from a controlled experiment, it is possible that they are explained by some systematic bias in the choice of three or four member groups. Porter et al. [55] compared one, two, and four person reviews and found that two reviewers outperformed one, but that four did not outperform two. Behavioral research offers a possible explanation in that it has found that expert pairs perform as well as larger groups [7]. A third set of results which bears on Proposition 6 is from the series of studies on N-fold inspections. These results are consistent with the prediction that performance increases with size, although the overall N-fold design is rather different from the SDTR task model [47], [60].

Proposition 7 says that, above a certain size of group, performance will start to decline. That is to say, process loss will exceed any gains from increased expertise. Experiments have found that SDTRs experience significant process loss [33], [34], [55], [82]. Unfortunately, it is difficult to assemble sufficient laboratory subjects to statistically test process loss across a range of different sizes. In the field, where suitable numbers of different sized reviews occur, researchers cannot readily determine whether issues discarded in the group meeting are true defects or false positives.

Formally, Proposition 8 is untested. Its contention that the distinctive contribution of a review meeting is to discriminate true defects from false positives is supported by the results of a laboratory experiment and its replication [34], [81]. These two studies found that the review meeting outperformed both average individual reviewers and the nominal group when false positives were offset against true defects. The probability of true defects identified by individuals being included in the group's report was significantly greater than the similar probability for false positives. Since these results come from a single experimental set-up, further studies by other researchers would be desirable. It is also relevant that Johnson and Tjahjono [29] found that interacting groups have fewer false positives than nominal groups.

Proposition 9's proposal that expert pairs are as good as larger groups at defect discrimination has not been formally tested. Porter et al.'s finding that pairs performed as well as groups of four [55] is particularly encouraging in that their study included the discovery stage, which might have been expected to obscure any differential performance in the discrimination task.

Proposition 10 implies that a nominal group design is the most effective for discovering defects. Laboratory and field research has explicitly demonstrated that nominal groups outperform both an average individual reviewer and the best member of the nominal group—many heads are, almost by definition, better than one [34], [82]. The option of using the review meeting as the principal defect discovery stage has been shown by other researchers not to outperform the nominal group design [29]. This is consistent with evidence from studies of brainstorming which find that, contrary to the widespread belief that more ideas are generated by an unprepared group, more ideas are, in fact, generated when individuals prepare in advance of the meeting [10]. In other words, if preparation pays even for brainstorming, it will be essential for effective defect discovery.

Proposition 11 restates Proposition 7 in the context of the nominal group design. Its implication is that the number of reviewers required to discover a given number of defects in a product depends on their expertise. As noted in our discussion of that proposition, this is difficult to study and currently relatively unstudied. The work most relevant is that based on N-fold inspection, which is a partial nominal design and which finds performance increases with scale [47], [60].

In summary, existing research on SDTRs already provides a degree of support for our proposed research program. Although few propositions have been explicitly tested, there is sufficient evidence of process loss and of synergy being insignificant to support our emphasis on defect discrimination as review meetings' principal contribution to effectiveness. The evidence in favor of nominal groups for defect discovery is also strong enough to justify our beginning to think about design alternatives based on the three tasks of discovery, collection, and discrimination.

5.2 Value of a Theoretically Motivated Research Program

This paper argues for a research program in software reviews which is structured around determinants of performance for the three component tasks of reviews: defect discovery, defect collection, and defect discrimination (Fig. 2).

The program is given coherence by the use of the behavioral theory to guide the development of a set of related propositions to advance our understanding of each task. It allows us to organize otherwise disparate research findings in a consistent and coherent manner which makes their importance clear (Table 3). This helps researchers make well-informed decisions about where to target their research. Because the propositions are theoretically related, research which tests them is likely to be cumulative. In addition, because the theory asserts the relative importance of factors, the program implicitly suggests research priorities. The program's implications for research are discussed in Section 5.2.1. The majority of propositions in the program (Propositions 5-11) are designed to test predictions of the

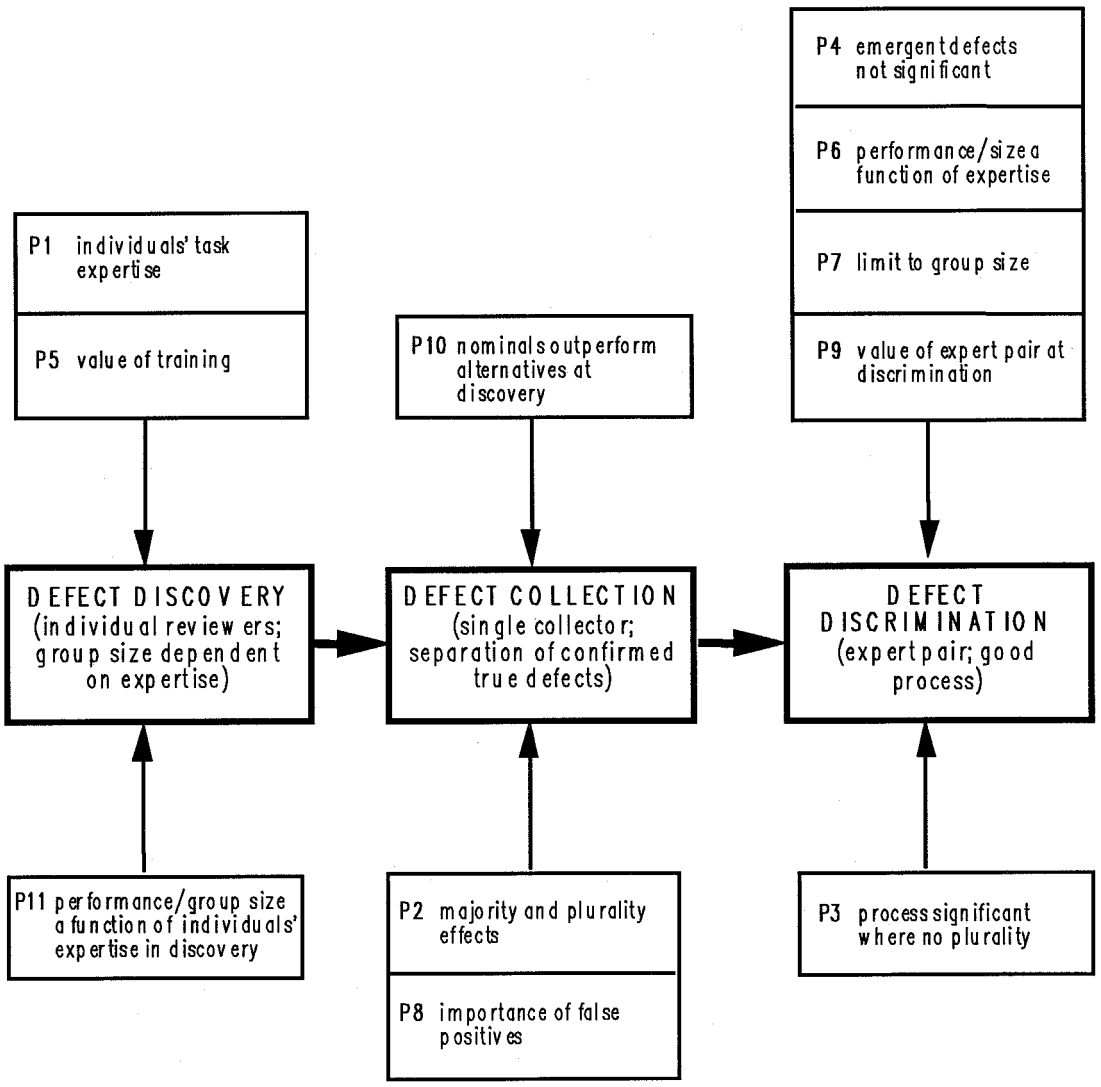

Fig. 2. Relationship of research propositions to SDTR component tasks.

underlying theory which have direct implications for practical interventions. The program's basis in an established theory means that there are rational grounds for exploring those interventions ahead of comprehensive testing. The program's implications for practice are discussed in Section 5.2.2.

5.2.1 Implications for Research

This paper's principal implication for research is the research program itself. It is one of the advantages of basing the program on an established theory that it also helps stimulate recognition of relevant research issues beyond the stated propositions.

It is a clear implication of our program that the first priority of future research should be the exploration and extension of our understanding of defect detection expertise. We need to understand what varieties of expertise there are among software engineers so as to understand the possible distribution of expertise, especially in defect discovery; what levels of expertise there are for the purposes of mapping performance/size functions; what are good predictors for selection purposes of individuals' expertise; and what developmental processes can enhance expertise. Existing research on expertise such as that on software reading [3], [4] must be expanded and accelerated to probe the essential skills of defect discovery and discrimination. The literature on understanding programs and on the psychology of debugging provides a source of knowledge about relevant expertise [9], [44], [52], [54], [64].

One of the strengths of the explanatory capacity of the underlying theory is the insight it gives us beyond the set of propositions formally constituting the program. For example, behavioral theory allows us to see that task-oriented meeting roles [78], review specialization [31], [53], and the scenario task aid [56] all work on the same principle, namely that defect detection is enhanced by specialization in the application of expertise. This immediately leads us to ask whether the source of any improvement is the need for different expertises or something quite different, such as the reduction in size of the individual tasks. The answer implies quite different interventions.

Using our program structure to organize existing research helps highlight issues with implications for our research methods. For example, the program's emphasis on

reviewers' task expertise raises the concern that experiments which use inexperienced subjects, such as undergraduate students, may produce results which are not applicable to industrial situations. Indeed, since expertise can be assumed to vary considerably across companies, the applicability of field-based results may also be constrained. As another example, in discussing in Section 5.1 how infrequently two reviewers discover the same defect, we speculated that discovery time might not be sufficient for the size of the product reviewed. We must ensure that our experiments are designed to avoid this validity threat. Otherwise, results may confound synergy with insufficient preparation time or excessive task size.

The underlying theory helps us to evaluate proposed interventions which are not part of the defined research program and to identify what knowledge from the program is relevant. For example, research on computer-supported reviews has, with exceptions [29], typically concentrated on demonstrating feasibility rather than on evaluating effectiveness. Behavioral theory permits us to make performance predictions relating to the typical features of computer-supported reviews which include document handling, task aids for individual preparation, meeting support, and data collection [45]. The theory predicts two potential sources of significant performance improvement and, hence, two focuses for research on computer support: computer-supported task aids, which enhance individual expertise more effectively than manual aids, and computer-imposed disciplines which prevent process loss.

Our program does not claim completeness. Our propositions need to be supplemented to advance our knowledge sufficiently to refine ideas for improving effectiveness. For example, we need to plot the different performance/size curves and limits on size for different levels of expertise before we can confidently prescribe review group sizes for specific situations. Also, we need to research a theory of defect detection. But, even the limits can have a creative dimension for researchers. For example, not knowing how common it is for multiple reviewers to find the same defect, we cannot know how important plurality effects are as a decision scheme for identifying expertise in a group. Evidence that plurality is uncommon suggests that there may be value in investigating whether there are alternative decision schemes used in group reviews. This is an entirely new question which may not have surfaced independent of this program. As empirical research develops, new avenues can be expected to appear.

5.2.2 Implications for Practice

Our program's basis in an established theory and the existence of some empirical support (Table 3) provide reasonable grounds for organizations to cautiously pilot either improvements to the current design (Section 5.2.2.1) or alternative designs (Section 5.2.2.2). The program also provides grounds for reviewing certain established management assumptions about reviews and introducing new, more effective management practices (Section 5.2.2.3).

Improvements to the Current Design. Performance under current review designs may be improved by selecting better reviewers, providing training, fine-tuning the size of reviews, and providing aids to expertise. Better selection implies testing defect detection skills when recruiting software engineers for roles which include software review. In-house, it implies measuring and monitoring individuals' review performance and using that as a guide for selection. Training requires us to understand defect detection expertise so that we can structure and transmit the necessary knowledge and skills. At present, we can only confidently provide generic problem-solving training. However, this should not be discounted as a real possibility for improving performance since it has been shown to be effective in behavioral research [6], [19], [80]. Performance can also be improved by increasing group size. Practitioners considering this option should monitor performance for different sized reviews to ensure that overall losses are not incurred. Finally, reading technologies, such as checklists and scenarios, which help enhance individuals' defect detection expertise, should be employed wherever possible [4].

Alternative Designs. Alternative review designs can be developed through separating the tasks of defect collection and defect discrimination. Careless collection arising from the distractions and constraints of group interaction causes loss of true defects between individuals' defect discovery and the report of the review group. Collection independent of defect discrimination only requires one person with a control process based on simple counts. Collection could also include identifying and marking for rework without further discussion any defects found by more than one reviewer. Introduction of such a task should eliminate collection losses and save review time.

Defect discrimination may either be discarded entirely or the number of reviewers may be reduced. If individual reviewers identify very few false positives or if false positives are cheaply and accurately identified by the author during rework, then there is no need to undertake defect discrimination. The collected issues identified by individuals can be passed direct to the author for rework. Indeed, the author might be assigned the task of collection as a prelude to rework. If the number of false positives proves significant, then defect discrimination can add value. Even so, the precedent of audit reviews suggests that one expert reviewer paired with the author may be sufficient to perform the discrimination task at least as accurately as a larger group.

Neither alternative is likely to be universally applicable. Organizations considering piloting an alternative can improve their likelihood of success by collecting data on the occurrence of false positives in their existing review process and on the review expertise of their software engineers. This would enable them to select the better design for them and to assign the most appropriate reviewer to the discrimination task. Of course, where no sufficiently expert reviewer is available, the more traditional design may still be the most effective.

The advantages of eliminating the defect discrimination task are reduced labor cost and reduced cycle-time. Defect discrimination by a single expert likewise has cost advantages and, because there is only the one reviewer to be scheduled, cycle time can be minimized. It might be possible to combine collection with expert discrimination,

but whether the two tasks would interfere with each other is a matter for empirical test.

The mere possibility of cutting back on or eliminating the review meeting is predicated on the assumption, supported by research, that emergent defects are numerically insignificant. However, it would be appropriate for organizations to check this before embarking on alternatives. If emergents do appear in significant numbers, they should also check whether the software product for review is too large to be adequately reviewed in the preparation time available and whether reviewers are not preparing thoroughly (free riders). In either case, the meeting serves as an extension of individual preparation time. It may, therefore, be more productive to make change to increase the probability of individual reviewers performing the discovery task exhaustively than to retain the review meeting.

Managing SDTRs. A major implication of the emphasis on individuals' expertise is that review management processes should change. In the past, the belief has been that review performance is collective, not individualistic. Our program suggests that individuals' expertise should be harnessed through performance measurement and the provision of incentives. This implies that it would now be appropriate to assess individuals' review performance in staff appraisals [66]. This is particularly important in respect to designs which increase the number of reviewers because they could encourage free rider behavior.

6 Conclusions

Behavioral theory helps us to answer the three questions with which we started:

1. What makes SDTRs effective at defect detection?
2. How can we improve SDTR performance within the current review design?
3. What design alternatives would theory predict to be more effective?

First, it tells us that the most important factor in determining the effectiveness of SDTRs is the level of expertise of the individual reviewers. Second, within the current two-stage design of SDTRs, it highlights three ways of improving performance: selection of reviewers who are expert at defect detection; training to improve individuals' expertise; and establishing group size at the limit of performance. Third, theory predicts that more effective designs may be achieved by taking best advantage of the relative merits of individuals and groups. Our current knowledge suggests a three stage design involving: 1) multiple individuals working independently to discover as many defects as possible, 2) a single collector, and 3) where false positives are significant, an expert pair discriminating true defects from false positives among those identified by just one reviewer. However, even within this structure, there remain several degrees of freedom for variant designs, the efficacy of which require further investigation.

The implications for current practice are that the best available reviewers should be used. While it may be possible to substitute numbers for expertise in defect discovery by employing more reviewers, behavioral theory suggests strongly that this would not work for a defect discrimination meeting. Where traditional group meetings are retained, care must be exercised not to increase group size beyond the limit of effectiveness. A balance needs to be struck between having more reviewers to increase defect discovery and having so many that some do not prepare adequately and the meeting experiences process loss. As our program suggests that there is no special significance in the interaction of the review group, it is now appropriate to apply normal management processes, including introducing incentives for improving defect detection performance and appraising individuals' actual performance against expectations.

In addition to the program of propositions to be validated, the implications for research are threefold. First, although behavioral theory does not consider problem-solving aids, its emphasis on expertise implies that current work on reading technologies has the potential to enhance review effectiveness. Second, defect detection expertise must be researched. Until we know what makes an expert reviewer, we will not be able to train effectively and training is the best way of improving performance across the industry. Third, a theory driven program is a powerful aid to research. It helps make research pertinent, structured, efficient, innovative and cumulative. Of course, our program is not exhaustive. Rather, it is a point of departure for further productive development of an already active area of research and practice.

Acknowledgments

The authors are grateful to Adam Porter for helpful comments on an earlier draft. The authors are also grateful to the associate editor and three anonymous referees for constructive suggestions for improving the paper.

References

[1] A.F. Ackerman, P.J. Fowler, and R. Ebenau, "Software Inspections and the Industrial Production of Software," *Software Validation,* North-Holland, 1984.

[2] ANSI/IEEE, *An American National Standard: IEEE Standard for Software Reviews and Audits,* ANSI/IEEE Std 1028-1988, New York: IEEE, 1989.

[3] V.R. Basili, G. Caldiera, F. Lanubile, and F. Shull, "Studies on Reading Techniques," *Proc. 21st Ann. Software Eng. Workshop,* Dec. 1996.

[4] V.R. Basili, S. Green, O. Laitenburger, F. Lanubile, F. Shull, S. Sorumgard, and M. Zelkowitz, "The Empirical Investigation of Perspective-Based Reading," *J. Empirical Software Eng.,* vol. 1, no. 2, 1996.

[5] D.B. Bisant and J.R. Lyle, "A Two-Person Inspection Method to Improve Programming Productivity," *IEEE Trans. Software Eng.,* vol. 15, no. 10, pp. 1,294-1,304, Oct. 1989.

[6] P.C. Bottger and P.W. Yetton, "Improving Group Performance by Training in Individual Problem Solving," *J. Applied Psychology,* vol. 72, no. 4, pp. 651-657, 1987.

[7] P.C. Bottger and P.W. Yetton, "An Integration of Process and Decision Scheme Explanations of Group Problem Solving Performance," *Organizational Behavior and Human Decision Processes,* vol. 42, pp. 234-249, 1988.

[8] M.G. Bradac, D.E. Perry, and L.G. Votta, "Prototyping a Process Monitoring Experiment," *Proc. 15th Int'l Conf. Software Eng.,* May 1993.

[9] T.A. Corbi, "Program Understanding: Challenge for the 1990s," *IBM Systems J.,* vol. 28, no. 2, pp. 294-306, 1989.

[10] A. Diehl and W. Stroebe, "Productivity Loss in Brainstorming Groups: Toward the Solution of a Riddle," *J. Personality and Social Psychology,* vol. 53, no. 3, pp. 497-509, 1987.

[11] S.G. Eick, C.R. Loader, M.D. Long, L.G. Votta, and S. Vander Wiel, "Estimating Fault Content before Coding," *Proc. 14th Int'l Conf. Software Eng.,* May 1992.

[12] H.J. Einhorn, R.M. Hogarth, and E. Klempner, "Quality of Group Judgement," *Psychological Bulletin,* vol. 84, pp. 158-172, 1977.

[13] M.E. Fagan, "Design and Code Inspections to Reduce Errors in Program Development," *IBM Systems J.,* vol. 15, no. 3, pp. 182-211, 1976.

[14] M.E. Fagan, "Advances in Software Inspections," *IEEE Trans. Software Eng.,* vol. 12, no. 7, pp. 744-751, July 1986.

[15] P.J. Fowler, "In-Process Inspections of Workproducts at AT&T," *AT&T Technical J.,* pp. 102-112, Mar./Apr. 1986.

[16] L.A. Franz and J.C. Shih, "Estimating the Value of Inspections for Early Testing of Software Projects," *Hewlett-Packard J.,* vol. 45, no. 6, Dec. 1994.

[17] D.P. Freedman and G.M. Weinberg, *Handbook of Walkthroughs, Inspections, and Technical Reviews: Evaluating Programs, Projects, and Products,* third ed. Little Brown & Co., 1982.

[18] R.D. Galliers, Y. Merali, and L. Spearing, "Managing Information Technology? How British Executives Perceive the Key Issues," *J. Information Technology,* vol. 9, no. 1, pp. 223-238, 1994.

[19] D.C. Ganster, P. Poppler, and S. Williams, "Does Training in Problem Solving Improve the Quality of Group Decisions?" *J. Applied Psychology,* vol. 76, no. 3, pp. 479-483, 1991.

[20] T. Gilb, *Principles of Software Engineering Management.* Wokingham: Addison-Wesley, 1988.

[21] R. Glazer, J.H. Steckel, and R.S. Winer, "Group Process and Decision Performance in a Simulated Marketing Environment," *J. Business Research,* vol. 15, pp. 545-557, 1987.

[22] R.B. Grady, *Practical Software Metrics for Project Management and Process Improvement.* Englewood Cliffs, N.J.: Prentice Hall, 1992.

[23] R.B. Grady and T. Van Slack, "Key Lessons in Achieving Widespread Inspection Use," *IEEE Software,* vol. 11, no. 4, July 1994.

[24] H.A. Gurnee, "A Comparison of Collective and Group Judgements of Fact," *J. Experimental Psychology,* vol. 3, pp. 437-444, 1937.

[25] J.R. Hackman and C.G. Morris, "Group Tasks, Group Interaction Process and Group Performance Effectiveness: A Review and Partial Integration," *Advances in Experimental Social Psychology,* L. Berkowitz, ed., vol. 8, pp. 47-99, New York: Academic Press, 1975.

[26] J. Hall and W.H. Watson, "The Effects of a Normative Intervention on Group Decision-Making Performance," *Human Relations,* vol. 23, pp. 299-317, 1970.

[27] R.A. Henry, "Improving Group Judgement Accuracy: Information Sharing and Determining the Best Member," *Organizational Behavior and Human Decision Processes,* vol. 62, no. 2, pp. 190-197, May 1995.

[28] C.R. Holloman and H.W. Hendrick, "Individual versus Group Effectiveness in Solving Factual and Nonfactual Problems," *Proc. 78th Ann. Convention, Am. Psychological Assoc.,* 1970.

[29] P.M. Johnson and D. Tjahjono, "Assessing Software Review Meetings: A Controlled Experimental Study Using CSRS," Technical Report 96-06, Dept. of Information and Computer Sciences, Univ. of Hawaii, 1996.

[30] L.P.W. Kim, C. Sauer, and R. Jeffery, "A Critical Survey of Software Development Technical Reviews as a Non-Method-Specific Approach to Software Quality Assurance," ITRC Report #94/34, School of Information Systems, Univ. of New South Wales, Sydney, 1994.

[31] J.C. Knight and A.N. Myers, "An Improved Inspection Technique," *Comm. ACM,* vol. 36, no. 11, pp. 51-61, Nov. 1993.

[32] I. Lakatos, "Falsification and the Methodology of Scientific Research Programmes," *Criticism and the Growth of Knowledge,* I. Lakatos and A. Musgrave, eds., Cambridge U.K.: Cambridge Univ. Press, 1974.

[33] F. Lanubile and G. Visaggio, "Assessing Defect Detection Methods for Software Requirements Inspections through External Replication," Technical Report ISERN-96-01, Int'l Software Eng. Research Network, Jan. 1996.

[34] L.P.W. Land, C. Sauer, and R. Jeffery, "Validating the Defect Detection Performance Advantage of Group Designs for Software Reviews: Report of a Laboratory Experiment Using Program Code," *Proc. Sixth European Software Eng. Conf.—ESEC/FSE '97,* pp. 294-309, 1997.

[35] P.R. Laughlin, N.L. Kerr, J.H. Davis, H.M. Halff, and K.A. Marcinak, "Group Size, Member Ability, and Social Decision Schemes on an Intellective Task," *J. Personality and Social Psychology,* vol. 31, no. 3, pp. 522-535, 1975.

[36] R.E. Levine and R.L. Moreland, "Progress in Small Group Research," *Ann. Review of Psychology,* vol. 1, 1990.

[37] R. Libby and R.K. Blashfield, "Performance of a Composite as a Function of the Number of Judges," *Organizational Behavior and Human Performance,* vol. 21, pp. 121-129, 1978.

[38] R. Libby, K.T. Trotman, and I. Zimmer, "Member Variation, Recognition of Expertise, and Group Performance" *J. Applied Psychology,* vol. 72, pp. 81-87, 1987.

[39] *Generalising from Laboratory to Field Settings: Research Findings from Industrial-Organizational Psychology, Organizational Behaviour, and Human Resource Management,* E.A. Locke, ed., Lexington, Mass.: Heath-Lexington, 1985.

[40] I. Lorge and H. Solomon, "Two Models of Group Behaviour in the Solution of Eureka Type Problems," *Psychometrica,* vol. 20, pp. 139-148, 1955.

[41] I. Lorge and H. Solomon, "Individual Performance and Group Performance in Problem Solving Related to Group Size and Previous Exposure to the Problem," *J. Psychology,* vol. 48, pp. 107-114, 1959.

[42] I. Lorge and H. Solomon, "Group and Individual Performance in Problem Solving Related to Previous Exposure to the Problem, Level of Aspiration, and Group Size," *Behavioral Science,* vol. 5, pp. 28-38, 1960.

[43] I. Lorge, D. Fox, J. Davitz, and M. Brenner, "A Survey of Studies Contrasting the Quality of Group Performance and Individual Performance," *Psychological Bulletin,* vol. 55, pp. 337-371, 1958.

[44] F.J. Lukey, "Understanding and Debugging Programs," *Int'l J. Man-Machine Studies,* vol. 12, pp. 189-202, 1980.

[45] F. Macdonald, J. Miller, A. Brooks, M. Roper, and M. Wood, "Automating the Software Inspection Process," *Proc. Seventh Int'l Workshop Computer-Aided Software Eng., CASE '95,* July 1995.

[46] R. Madachy, L. Little, and S. Fan, "Analysis of a Successful Inspection Program," *Proc. Software Eng. Workshop, SEW,* SEL-93-003, pp. 176-188, 1993.

[47] J. Martin and W.T. Tsai, "N-fold Inspection: A Requirements Analysis Technique," *Comm. ACM,* vol. 33, no. 2, pp. 225-232, Feb. 1990.

[48] V. Mashayekhi, J.M. Drake, W.T. Tsai, and J. Riedl, "Distributed, Collaborative Software Inspections," *IEEE Software,* pp. 66-75, Sept. 1993.

[49] L.K. Michaelsen, W.E. Watson, and R.H. Black, "Realistic Test of Individual versus Group Decision Making," *J. Applied Psychology,* vol. 74, pp. 834-839, 1989.

[50] G.J. Myers, "A Controlled Experiment in Program Testing and Code Walkthroughs/Inspections," *Comm. ACM,* vol. 21, no. 9, pp. 760-768, Sept. 1978.

[51] F. Niederman, J.C. Brancheau, and J.C. Wetherbe, "Information Systems Management Issues for the 1990s," *MIS Quarterly,* pp. 475-500, Dec. 1991.

[52] G.M. Olson, S. Sheppard, and E. Soloway, *Empirical Studies of Programmers: Second Workshop.* Norwood, N.J.: Ablex Publishing, 1987.

[53] D.L. Parnas and D.M. Weiss, "Active Design Reviews: Principles and Practices," *Proc. Eighth Int'l Conf. Software Eng.,* pp. 215-222, Aug. 1985.

[54] N. Pennington, "Stimulus Structures and Mental Representations in Expert Comprehension of Computer Programs," *Cognitive Psychology,* vol. 19, pp. 295-341, 1987.

[55] A.A. Porter, H. Siy, C.A. Toman, and L.G. Votta, "An Experiment to Assess the Cost-Benefits of Code Inspections in Large-Scale Software Development," http://www.cs.umd.edu/~aporter/live.ps., 1996.

[56] A.A. Porter and L.G. Votta, "An Experiment to Assess Different Defect Detection Methods for Software Requirements Inspections," *Proc. 16th Int'l Conf. Software Eng.,* pp. 103-112, May 1994.

[57] A.A. Porter, L.G. Votta, and V.R. Basili, "Comparing Detection Methods for Software Requirements Inspections: A Replicated Experiment," *IEEE Trans. Software Eng.,* vol. 21, no. 6, pp. 563-575, June 1995.

[58] Z.W. Pylyshyn, *Computation and Cognition: Toward a Foundation for Cognitive Science.* Cambridge, Mass: Bradford Books, MIT Press, 1984.

[59] F. Redmill, "Fagan's Inspection: Achieving Quality in Code and Documentation; Built-In Gauge of Effectiveness," *Managing System Development,* vol. 13, no. 3, pp. 1-5, Mar. 1993.
[60] G.M. Schneider, J. Martin, and W.T. Tsai, "An Experimental Study of Fault Detection in User Requirements Documents," *ACM Trans. Software Eng. and Methodology,* vol. 1, no. 2, pp. 188-204, Apr. 1992.
[61] M.E. Shaw, "Comparison of Individuals and Small Groups in the Rational Solution of Complex Problems," *Am. J. Psychology,* vol. 44, pp. 491-504, 1932.
[62] M.E. Shaw, *Group Dynamics: The Psychology of Small Group Behavior.* New York: McGraw-Hill, 1981.
[63] H.P. Siy, "Identifying the Mechanisms Driving Code Inspection Costs and Benefits," PhD thesis, Univ. of Maryland, 1996 (unpublished).
[64] *Empirical Studies of Programmers,* E. Soloway and S. Iyengar, eds. Norwood, N.J.: Ablex Publishing, 1986.
[65] I.D. Steiner, *Group Process and Productivity.* New York: Academic Press, 1972.
[66] R.G. Strauss and R.G. Ebenau, *Software Inspection Process.* New York: McGraw-Hill, 1994.
[67] R.L. Thorndike, "In What Type of Task Does the Group Do Well?" *J. Abnormal and Social Psychology,* vol. 33, pp. 408-412, 1938.
[68] K.T. Trotman, "The Review Process and the Accuracy of Auditor Judgements," *J. Accounting Research,* vol. 23, no. 2, pp. 740-752, 1985.
[69] K.T. Trotman and P.W. Yetton, "The Effect of the Review Process on Auditor Judgements," *J. Accounting Research,* vol. 23, no. 1, pp. 256-267, 1985.
[70] K.T. Trotman, P.W. Yetton, and I.R. Zimmer, "Group Size and Performance: Prediction of Failure by Loan Officers," *Australian J. Management,* pp. 127-136, 1981.
[71] K.T. Trotman, P.W. Yetton, and I.R. Zimmer, "Individual and Group Judgements of Internal Control Systems," *J. Accounting Research,* vol. 21, no. 1, pp. 286-292, 1983.
[72] L.G. Votta, "Does Every Inspection Need a Meeting?" *Proc. First ACM SIGSOFT Symp. Foundations of Software Eng., Software Eng. Notes,* D. Notkin, ed., vol. 18, no. 5, pp. 107-114, Dec. 1993.
[73] W. Watson, W. Sharp, and L.K. Michaelsen, "Member Competence, Group Interaction, and Group Decision Making: A Longitudinal Study," *J. Applied Psychology,* vol. 76, no. 6, pp. 803-809, 1991.
[74] G.M. Weinberg and D.P. Freedman, "Reviews, Walkthroughs, and Inspections," *IEEE Trans. Software Eng.,* vol. 10, no. 1, pp. 68-72, Jan. 1984.
[75] E.F. Weller, "Lessons from Three Years of Inspection Data," *IEEE Software,* pp. 38-45, Sept. 1993.
[76] P.W. Yetton and P.C. Bottger, "Individual versus Group Problem Solving: An Empirical Test of a Best-Member Strategy," *Organizational Behavior and Human Performance,* vol. 29, no. 3, pp. 307-321, June 1982.
[77] P.W. Yetton and P.C. Bottger, "The Relationship among Group Size, Member Ability, Social Decision Schemes, and Performance," *Organizational Behavior and Human Performance,* vol. 32, pp. 145-159, 1983.
[78] E. Yourdon, *Structured Walkthroughs,* fourth ed. Prentice Hall, 1989.
[79] R.C. Ziller, "Group Size: A Determinant of the Quality and Stability of Group Decisions," *Sociometry,* vol. 20, pp. 165-173, 1957.
[80] S. Rifkin and L. Deimel, "Applying Program Comprehension Techniques to Improve Software Inspections," *Proc. 19th Ann. NASA Software Laboratory Workshop,* Nov. 1994.
[81] L. Land, C. Sauer, and R. Jeffery, "The Performance Effects of Process Roles in Code Reviews: A Preliminary Empirical Investigation," Caesar Technical Report 97/7, Univ. of New South Wales, School of Information Systems, Sydney, 1997.
[82] L.P.W. Land, R. Jeffery, and C. Sauer, "Validating the Defect Detection Performance Advantage of Group Designs for Software Reviews: Report of a Replicated Experiment," *Proc. Australian Software Eng. Conf.,* P.A. Bailes ed., 1997.
[83] S.H. Kan, V.R. Basili, and L.N. Shapiro, "Software Quality: An Overview from the Perspective of Total Quality Management," *IBM Systems J.,* vol. 33, no. 1, pp. 4-18, 1994.
[84] E. Yourdon, "Quality: What It Means and How to Achieve It," *Management Information Science,* pp. 43-47, Feb. 1993.
[85] M.C. Paulk, B. Vurtis, and M.B. Chrissis, "Capability Maturity Model, version 1.1," *IEEE Software,* pp. 18-27, July 1993.
[86] A.K. Onoma and T. Yamaura, "Practical Steps Toward Quality Development," *IEEE Software,* pp. 68-76, Sept. 1995.
[87] P. Hsia, "Learning to Put Lessons into Practice," *IEEE Software,* pp. 14-17, Sept. 1993.
[88] T. Gilb and D. Graham, *Software Inspection.* Harlow, Essex: Addison-Wesley, 1993.

Chris Sauer is a graduate of Oxford University and was awarded his doctorate in management from the University of Western Australia. He is a research fellow in information management at the Oxford Institute of Information Management at Templeton College, Oxford University. He performs research on general management issues in information technology (IT). His latest book, *Steps to the Future: Fresh Thinking in the Management of IT-Based Organizational Transformation*, coauthored with Philip Yetton and associates, has been published by Jossey-Bass, 1997. He holds several positions with academic journals and is secretary to the IFIP Working Group 8.6 on Information Technology Diffusion, Transfer, and Implementation.

D. Ross Jeffery is a professor of information systems at the University of New South Wales and director of the Centre for Advanced Empirical Software Research, Sydney. His research interests are in the areas of software metrics, cost estimation, software inspections, and software engineering management. He is on the editorial board of the journal *IEEE Transactions on Software Engineering* and also an associate editor for *Empirical Software Engineering: An International Journal.* He has published numerous books and articles in recent years and acted as a consultant to many Australian, as well as European, companies.

Lesley Land received her BSc degree (honors) in computer science from the University College London, University of London, and her MSc degree in intelligent systems from Brunel University, United Kingdom. She is an assistant lecturer and PhD candidate in the School of Information Systems at the University of New South Wales, Sydney. She has previously worked in the School of Computer Science and Engineering at the University of New South Wales and in the Department of Computer Science at the University of Western Australia. Her research interests include software inspections and software development process improvement; she has published several papers on these topics.

Philip Yetton is a graduate of Cambridge, Liverpool, and Carnegie-Mellon Universities. He is now the Commonwealth Rank Professor of Management and the executive director of the Fujitsu Centre for Managing Information Technology in Organizations at the Australian Graduate School of Management, Sydney, where he teaches the management of information technology (IT), strategic leadership, and organizational behavior. His major research interests are in information technology, leadership style, and decision making. He has contributed major research advances to organizational behavior, including leadership theory and the behavioral theory of group performance. He has written numerous research papers and is on the editorial board of *Leadership Quarterly* and *Organizational Science*.