Evaluating Guidelines for Empirical Software Engineering Studies

Barbara Kitchenham[1], Hiyam Al-Khilidar[1,2], Muhammad Ali Babar[1,2], Mike Berry[1,2], Karl Cox[1], Jacky Keung[1,2], Felicia Kurniawati[1], Mark Staples[1,2], He Zhang[1,2], Liming Zhu[1]

[1] National ICT Australia Ltd; Australian Technology Park, Sydney 1430 NSW, Australia;
{barbara.kitchenham, hiyam.al-kilidar, muhammad.alibabar, mike.berry, karl.cox, jacky.keung, felicia.kurniawata, mark.staples, he.zhang, liming.zhu}@nicta.com.au
[2]School of Computer Science & Engineering, University of New South Wales, Sydney 2052 NSW, Australia

ABSTRACT

Background. Several researchers have criticized the standards of performing and reporting empirical studies in software engineering. In order to address this problem, Andreas Jedlitschka and Dietmar Pfahl have produced reporting guidelines for controlled experiments in software engineering. They pointed out that their guidelines needed evaluation. We agree that guidelines need to be evaluated before they can be widely adopted. If guidelines are flawed, they will cause more problems that they solve.

Aim. The aim of this paper is to present the method we used to evaluate the guidelines and report the results of our evaluation exercise. We suggest our evaluation process may be of more general use if reporting guidelines for other types of empirical study are developed.

Method. We used perspective-based inspections to perform a theoretical evaluation of the guidelines. A separate inspection was performed for each perspective. The perspectives used were: Researcher, Practitioner/Consultant, Meta-analyst, Replicator, Reviewer and Author. Apart from the Author perspective, the inspections were based on a set of questions derived by brainstorming. The inspection using the Author perspective reviewed each section of the guidelines sequentially.

Results. The question-based perspective inspections detected 42 issues where the guidelines would benefit from amendment or clarification and 8 defects.

Conclusions. Reporting guidelines need to specify what information goes into what section and avoid excessive duplication. Software engineering researchers need to be cautious about adopting reporting guidelines that differ from those used by other disciplines. The current guidelines need to be revised and the revised guidelines need to be subjected to further theoretical and empirical validation. Perspective-based inspection is a useful validation method but the practitioner/consultant perspective presents difficulties.

ISESE'06, September 21–22, 2006, Rio de Janeiro, Brazil.

Categories and Subject Descriptors

K.6.3 [**Software Engineering**]: Software Management – Software process.

General Terms: Management, Experimentation.

Keywords: Controlled experiments, software engineering, guidelines, perspective-based inspection

1. INTRODUCTION

This paper reports an exercise undertaken by staff and students in the Empirical Software Engineering (ESE) group at the National ICT Australia to evaluate the reporting guidelines for controlled experiments proposed by Jedlitschka and Pfahl [6]. In spite of the existence of a specialist book to help software engineers conduct experiments [14], software engineering experiments are still subject to criticism. The guidelines were developed in response to general criticisms of current standards of performing and reporting empirical studies (Kitchenham et al. [7]), and more specific criticisms that the lack of reporting standards is causing problems when researchers attempt to aggregate empirical evidence because important information is not reported or is reported in an inconsistent fashion (e.g. [9], [13]).

In fact, controlled experiments are performed infrequently in software engineering. A recent survey of 5,453 software engineering articles from 12 leading conference and journals that found only 103 articles that could be categorized as experiments [11]. However, there is evidence that current reporting practice is inadequate. Dybå et al. [3] had to exclude 21[1] experiments from their analysis of power because they did not report enough information for a power analysis. 14 experiments did not perform any statistical analysis and 7 experiments were so badly documented that Dybå et al. "did not manage to track which tests answered which hypothesis or research question". This result confirms the need for reporting guidelines for software engineering experiments.

Our evaluation is a response to Jedlitschka and Pfahl's recognition that their guidelines need to be evaluated:

[1] Another 5 experiments were excluded because they were reported in more than one article.

"Our proposal has not yet been evaluated e.g. through peer review by stakeholders, or by applying it to a number of significant number of controlled experiments to check its usability. We are aware that this proposal can only be the first step towards a standardized reporting guideline." [6]

We agree with the need for guidelines to be evaluated. If the guidelines are themselves flawed, they could make the problem worse that it is currently.

Our evaluation exercise took place between 5th October 2005 and December 14th 2005. It was organized as a series of 8 working meetings each taking between 1 and 2.5 hours. In this paper, we report the evaluation method we used and the results of our evaluation. We have already reported our results to Jedlitschka and Pfahl, so the main purpose of this paper is to report our evaluation method, since it might prove useful to other groups wanting to evaluate the next version of experimental guidelines or future reporting guidelines for other forms of empirical study such as case studies, surveys, or systematic reviews.

In section 2 we give a brief overview of the proposed guidelines. In Section 3 we discuss the various options available for evaluating experimental guidelines and provide a rational for our choice of perspective-based inspections. In Section 4 we report our evaluation process. In section 5 we report our evaluation results. In section 5 we discuss our results.

2. PROPOSED REPORTING GUIDELINES

Jedlitschka and Pfahl [6] propose a reporting structure based on the following sections (the reference in parenthesis is the section in [6] that explains the meaning of the guidelines section):

Title

Authorship

Structured Abstract (3.1)

Motivation (3.2) with subsections

- Problem Statement (3.2.1)
- Research Objectives (3.2.3), in the formalized style used in GQM
- Context (3.2.3), environmental factors

Related work (3.3), how current study fits in

Experimental design (3.4) with subsections:

- Goals, Hypotheses and Variables (3.4.1), formalization
- Design (3.4.2), which type
- Subjects (3.4.3), population sampling and group allocation
- Objects (3.4.4), what and why selected
- Instrumentation (3.4.5), material, tools and how used
- Data Collection Procedure (3.4.6)
- Analysis Procedure (3.4.7)
- Evaluation of Validity (3.4.8), how validity is assured.

Execution (3.5) with subsections:

- Sample (3.5.1), what does it look like
- Preparation (3.5.2), for experiment execution
- Data Collection Performed (3.5.3), actual process used
- Validity Procedure (3.5.4), how was data validated

Analysis (3.6) with subsections:

- Descriptive statistics (3.6.1), results
- Data set reduction (3.6.2), why and how
- Hypothesis testing (3.6.3), how analysis model and data were validated

Interpretation (3/7) with subsections

- Evaluation of results and implications (3.7.1)
- Limitations of Study (3.7.2), i.e. validity threats
- Inferences (3.7.3), i.e. generalizations
- Lesson learnt (3.7.4), experiences collected during experiment

Conclusions and Future work (3.8), with subsections (3.8)

- Relation to Existing Evidence (3.81), in context of section 3.
- Impact (3.8.2), cost, quality, schedule
- Limitations (3.8.3) when benefits will not be delivered
- Future work (3.8.4)

Acknowledgements (3.10)

References (3.11)

Appendices (3.12)

3. EVALUATION OPTIONS

At our first working meeting, we discussed various theoretical and empirical evaluation methods and considered the viability of each.

Theoretical evaluation could be based on several different approaches:

1. An assessment of each element in the guidelines from the viewpoint of why the element is included in the guidelines and what it is intended to accomplish in terms of supporting readers to find the information they are looking for, understand the experiment's conduct and assess the validity of its results, and what evidence there is to support the view that the element is important.
2. A review of the process by which the guidelines were constructed identifying the validity of the source material and the aggregation process.
3. An inspection of the guidelines by software engineering researchers taking a variety of perspectives and assessing how well the guidelines support each perspective.
4. Mapping from established experimental methodology guidelines to reporting guidelines.

Empirical evaluation could be based on a variety of possible approaches, for example:

1. Take a sample of experimental papers constructed without support of the guidelines and identify whether important information has been omitted from the papers that would have been included if the guidelines had been followed. This is similar to the approach taken by the CONSORT who compared papers in journals that used CONSORT with those that did not [8]. The objection to this approach is that the guidelines being evaluated are the basis for their own evaluation.
2. Take a sample of published experiments and re-structure them to conform to the guidelines and use the duplicate versions as the experimental material in an experiment aimed at evaluating whether the guidelines make it easier a) to

understand the papers and/or b) to extract standard information form the papers.

When deciding which evaluation process to undertake, we considered:

1. Whether the evaluation approach itself was valid i.e. likely to lead to a trustworthy assessment of the strengths and weaknesses of the guidelines.
2. Whether the evaluation approach was feasible given our resources (effort, time and people).
3. Whether the approach was cost effective given the value of the proposed guidelines. We noted that formal experiments are currently not often used in software engineering research. It is possible that effort spent on guidelines for industry case studies and surveys might be more beneficial.
4. Whether the approach provided a good learning opportunity for our research group as a whole. This was an important issue because the group included PhD students who were learning about empirical software engineering as well as more experienced researchers.

Table 1 Assessment of Evaluation methods. The type of approach: Theoretical (T) or Empirical (E).

Evaluation method	Type	Valid	Feasible for us	Cost effective	Learning potential
Evaluation of each element	T	Yes	Yes	Yes	No
Evaluation of process	T	Yes	Yes	Yes	No
Perspective -based inspections	T	Yes	Yes	Yes	Yes
Mapping to existing guidelines	T	Yes	Yes	Yes	No
Review of existing papers	E	No	Yes	No	Yes
Formal experiment	E	Yes	No	No	No

After evaluating each approach, as summarized in Table 1, we concluded that perspective-based inspection would be the most appropriate evaluation method for us to undertake. We felt that empirical evaluation was extremely problematic. In one case it was infeasible (the guidelines are not currently in use so we could not compare papers that used the guidelines with those that did not) and perhaps invalid, and in the other case it was too difficult (it would be difficult to reconfigure existing papers, particularly if information required by the standards was not available in the original papers). Of the theoretical evaluation methods, we felt the perspective-based inspection approach would provide the best learning opportunity for the junior researchers, giving them an opportunity to consider the needs of different readers and discuss, with more experienced researchers, .how to met those needs

4. APPLYING PERSPECTIVE-BASED INSPECTIONS TO EVALUATING THE EXPERIMENTAL GUIDELINES

4.1 Evaluation Process

Our evaluation process was organized as a series of 8 meetings each of which took between 1 and 2.5 hours. The date of each meeting, its purpose and outcome is reported in Table 2. The results of each meeting were documented after each meeting.

Table 2 Evaluation Process

#	Date	Purpose	Outcome
1	5th Oct.	Gain agreement on the evaluation method/approach.	Agreed to perform a perspective-based inspection. Identified the perspectives we would use.
2	12th Oct.	Specify a perspective-based checklist for researchers and practitioner/consultants.	Checklist questions identified.
3	19th Oct.	Review the guidelines from the perspective of researchers.	A list of problems and defects.
4	26th Oct.	Review the guidelines from the perspective of practitioner/consultants.	A list of problems and defects.
5	2nd Nov.	Review progress and decide whether to continue with remaining perspectives.	Developed checklist questions for meta-analyst, replicator and reviewer. Decided to treat the author perspective differently.
6	23rd Nov.	Review from the meta-analyst perspective.	A list of problems and defects.
7	7th Dec.	Review from the perspective of a replicator and a reviewer.	Two lists of problems and defects.
8	14th Dec.	Review from the perspective of an author.	A list of problems and defects.

4.2 Identification of the Relevant Perspectives

We identified the following perspectives of interest:

- **Researcher** who reads a paper to discover whether it offers important new information on a topic area that concerns him or her.

- **Practitioner/consultant** who provides summary information for use in industry and wants to know whether the results in the paper are likely to be of value to his/her company or clients.
- **Meta-analyst** who reads a paper in order to extract quantitative information that can be integrated with results of other equivalent experiments.
- **Replicator** who reads a paper with the aim of repeating the experiment.
- **Reviewer** who reads a paper on behalf of a journal or conference to ensure that it is suitable for publication.
- **Author** who would be expected to use the guidelines directly to report his/her experiment.

We also identified the perspective of the editorial board of journals (or the program committee of conferences) who might choose to adopt experimental guidelines. The adoption or not of a set of international guidelines could have both good and bad impacts:

- It might suggest to authors that there is a fast track to publication or acceptance by using the guidelines irrespective of the quality of the paper.
- It might discourage authors of non-experimental studies from submitting to the journal.
- It might improve the quality of papers.
- It might improve the quality of reviews.

However, although we believe the perspective of an editorial board is important we did not think it would be one that we could realistically adopt.

For each perspective, we used brainstorming to assess what an individual with each perspective would require from a paper and converted these issues into a number of questions that summarize the issues of importance to each perspective. The checklists we developed for Researcher, Practitioner/Consultants, Meta-Analysts, Replicator and Reviewer are shown in Table 3, Table 4, Table 5, Table 6, and Table 7 respectively. For the Researcher and Practitioner/Consultant perspective we did not attempt to remove duplicate questions thinking that it was important to fully represent each perspective. After applying both of these perspectives, we developed the Meta-analyst, Replicator and Reviewer perspectives. For these perspectives, we concentrated on the main differences between each perspective and the Researcher and Practitioner/Consultant perspectives. After our experience with the first two perspectives, we realized that there would be too much redundancy in the questions if we produced a complete checklist for each perspective.

We also decided not to attempt to construct a checklist for the Author perspective since it would be too close to the Researcher perspective. Instead we decided to undertake a separate inspection of the guidelines where we considered each element in turn discussing whether:

- Including the information would be difficult for authors.
- The guideline element was necessary.
- Including the information would improve the paper.
- Including the information would make the paper more difficult to read or write.

Using a different approach for reviewing from the author perspective gave us the chance to address issues not raised explicitly by brainstorming sessions.

Table 3 Researcher checklist

Number	Question	Rationale
1	Is the paper easy to find?	It is important for researchers to be able to find relevant research results.
2	Is it a relevant paper?	Having found a paper, a researcher should be able to identify quickly whether or not it is relevant to his/her research.
3	Is the overall structure of the paper appropriate?	Researchers need to be able to find easily specific pieces of information within a paper.
4	Is the research problem hypothesis easy to identify?	Researchers need to be sure exactly what the hypothesis the experiment is testing.
5	Is there an underlying causal model? If so, what is it?	It is important to know whether the research was derived from an underlying model and what it is.
6	Is the terminology defined and explained?	All specialized terminology needs to be defined.
7	Is the level of assumed knowledge excessive?	Junior researchers and researchers from other fields need sufficient explanation to follow the paper, or at least need to be directed to text books or reference articles where they can obtain information.
8	Is required background knowledge referenced?	
9	Is the research related to other relevant research?	Researchers need to know what the state of knowledge was prior to the experiment and how the current experiment contributes to new knowledge.
10	Is the experimental design appropriate?	Researchers need to know whether the experiment was capable of properly testing the hypothesis.

11	Is the statistical analysis correct?	Researchers need to be sure that the analysis was performed correctly.
12	Is the raw data available?	Researchers should be able to replicate the analysis or investigate alternative analysis methods. In order to do that the raw data should either be published in the paper or stated to be available on request.
13	Is it easy to identify the findings / results of the experiment?	Researchers need to know what the results of the experiment were.
14	Do the conclusions arise from the results?	Researchers need to be sure that the conclusions arise from the reported research results.
15	Is the argumentation clear?	Researchers need to be sure that any claims made in the paper (such as generalizations) are clearly linked to evidence which supports those claims.
16	Are limitations of the experiment made clear?	Researchers need to know the limitations, risks and constraints that apply to the experiment and the conclusions.
17	Is there any discussion of required further research?	Researches need to know what still needs to be investigated.

Table 4 Practitioner/Consultant checklist

Number	Question	Rationale
1	Is the paper easy to find?	It is important for consultants to be able to find relevant research results.
2	Is it a relevant paper?	Having found a paper, a consultant should be able to identify quickly whether or not it is relevant to his/her requirements.
3	What does the paper claim?	Consultants need to identify exactly what claims the paper makes about the technology of interest.
4	Are the conclusions/results useful?	Consultants need to be able to assess whether the conclusions/results are likely to have practical relevance.
5	Is the claim supported by believable evidence?	Consultants need to be sure that the claims made about the approach / method / technology are supported by the evidence provided.
6	Is it clear how the current research relates to existing research topics and trends?	Consultants need to know how the current work relates to existing research trends.
7	How can the results be used in practice?	Consultants need guidance on how the results would be used in industry.
8	In what context is the result/claim useful/relevant?	Consultants needs to know the context in which the results are expected to be use/useful.
9	Is the application type specified?	Consultants need to know what type of applications the results apply to. In particular whether they are specific to particular types of application (e.g. finance, or command and control etc.).
10	Is the availability of required support environment clear?	If the approach requires tool support, consultants need to know whether the tool is available and under what conditions.
11	Are any technology pre-requisites specified?	Consultants need to know whether there are any technological prerequisites that might limit the applicability of the results.
12	Are the experience or training costs required by development staff defined?	Consultants need to know the training/experience requirements implicit in the approach.
13	Is the expense involved in adopting the approach defined?	Consultants need some idea of the cost of adopting the approach, in order to perform return on investment (ROI) analyses.
14	Are any risks associated with adoption defined?	Consultants need to know whether there are any risks associated with adoption of the technique.
15	Do the results scale to real life?	Consultants need to be sure that the results scale to real life.
16	Is the experiment based on concrete examples of use/application or only theoretical models?	Consultants need to be sure that the results have a clear practical application.

17	Does the paper discuss existing technologies, in particular the technologies it supersedes and the technologies it builds on?	Consultants need to be sure that the experiment involves comparisons of appropriate technologies. They need to know that a new approach is better than other equivalent approaches not a "straw man".
18	Is new approach/technique/technology well described?	Consultants must be sure that they understand the new approach/technique/technology well enough to be able to adopt it.
19	Does the paper make it clear who is funding the experiment and whether they have any vested interests?	Consultants need to be sure that the experiment is as objective as possible.
20	Does the paper make it clear what commitment is required to adopt the technology?	A consultant needs to know whether adoption of an approach/technology requires a complete and radical process change or can be introduced incrementally.
21	Are Technology Transfer issues discussed?	Consultants need to know what the objections to a new technology are likely to be. Also whether there are any clear motivators or de-motivators
22	Is there any discussion of required further research?	Consultants need to know whether the result is complete or the approach needs further development.

Table 5 Meta-analyst Perspective

Number	Question	Rationale
1	How many experimental units per treatment?	The number of experimental units (subjects) is critical for meta-analysis.
2	What was effect size (or mean effect for each treatment and the variance)?	The effect size (or mean treatment effects and variance) are the basic data required for meta analysis.
3	Are treatments/technologies clearly defined?	The meta analyst must ensure that information from different studies pertains to the same treatments so that it can be aggregated.
4	Are the measures properly defined?	It is important to be sure that the measures used in different papers are equivalent.
5	Is the data collection process reliable?	It is important to be sure that the measurement collection follows a rigorous process.
6	Is the experimental procedure well defined?	It is important to ensure that experimental procedures are equivalent in different papers
7	Does the data analysis method match the stated experimental design?	It is important that the analysis results are correct.
8	Are any data transformation or reduction processes reported?	A meta-analyst needs to know if and how the data has been manipulated before analysis.
9	How are drop outs analyzed?	Differential drop-outs can seriously bias experimental results. The analysis protocol needs to address how drops out were handled.
10	Were experimental units allocated at random to treatment conditions?	Random allocation is a basic requirement for a randomized controlled experiment (as opposed to a quasi-random experiment).
11	Was the random allocation process defined?	Unless the randomization process is reported it cannot be assumed that random allocation (as opposed to haphazard allocation) has taken place.
12	Was sensitivity analysis performed?	The meta analyst needs to know that the results are robust (i.e. not the result of one or two atypical values.)
13	Was any form of blinding used?	Blinding is an essential means of reducing experimenter expectation bias. Opportunities are limited in software engineering experiments but it is sometimes possible to perform blind marking, and/or blind allocation to treatment. It is also possible to perform blind analysis (treatments are coded before data are given to the analyst).
14	Are any side-effects, or risks associated with the treatments defined?	It is important to be sure that any risks associated with new treatments are reported.

Table 6 Replicator Perspective

Number	Question	Rationale
1	Can I contact the authors if there are ambiguities in the description of the experiment?	Replicators need to be able to contact the experimenters if details are missing.[2]
2	Are the hypothesis fully defined?	Replicators may (and perhaps should) change the details of the experimental protocol. However, they must keep the same hypotheses (or they are not performing a replication).
3	Are subject groups clearly defined?	Whether the replicator wants to use different subjects or replicate with the same type of subject, he/she needs to know what sort of subjects were used in the first experiment.
4	Is it clear how the method/technology works including all necessary assumptions?	The replicator need to understand the technologies / methods being evaluated in order to construct test materials and devise test tasks.
5	Is the conduct of the experiment clearly defined?	The replicator must know how the experiment was performed in order to replicate it.
6	Are any problems or difficulties associated with the experimental protocol identified?	The replicator needs to know if there are any issues with the experimental protocol that need to be improved in a replication.
7	Is the effect size reported for power analysis?	A replicator should be able to perform a power analysis to determine the required number of experimental units.
8	Are the training requirements for subjects clear?	The replicator needs to provide appropriate training for subjects for all treatment conditions.
9	Are experimental materials available for consultation?	The replicator may need to consult the experimental materials used by the original experimenters.

Table 7 Reviewer's Perspective

Number	Question	Rationale
1	Is the paper original?	The first priority for a reviewer is to establish that the paper is neither plagiarized nor a copy of a previously published paper.
2	What is the contribution of the paper	The reviewer needs to assess whether the contribution of the paper is sufficient to warrant publication.
3	Are the references appropriate?	A reviewer needs to assess whether the authors has an appropriate knowledge of the field. Note the requirement of some journals for citation-based references seriously hinders the location of specific references.
4	Is background work cited?	Related to the issue of references, a reviewers need to assess whether all relevant background material is properly cited.
5	Is the design correct?	Reviews need to assess whether the design is appropriate to test the stated hypotheses.
6	Is the analysis correct?	Reviewers need to confirm that the analysis is consistent with the specified design.
7	Is it readable to the intended audience?	Given the audience of the journal or the expected background of conference participants, is the language used in the paper appropriate.

4.3 Performing Inspection

Unlike a conventional perspective-based inspection, we did not restrict each person to a single perspective. We performed an inspection for each perspective. We chose this approach because of the learning opportunities implicit in this process. Assigning individual perspectives to the inspection team would have been more efficient but it may not have lead to an inspection with the same level of scrutiny given to each perspective.

For the first two inspections, in order to assist us to understand each perspective, we agreed to read a paper reporting an experiment from International Symposium on Empirical Software Engineering (ISESE) 04 at the same time as we read the guidelines. Four of the 26 papers in the ISESE 04 conference proceedings reported experiments ([1], [2], [10], [12]) and each member of the group chose one of the papers to help with the inspection process. The choice of paper was not mandated and most people chose to read [12], while no one opted for [1]. We

[2] This is also important for meta analysts.

note that the relatively small number of experiments in a conference specializing in empirical methods confirms that experiments are currently not a major part of empirical software engineering.

Table 8 Inspection perspective and paper selection

Person	Perspective	Paper
Hiyam Al-Kilidar	Researcher	[12]
Muhammad Ali Babar	Practitioner	[12]
Mike Berry	Practitioner	[2]
Karl Cox	Practitioner	[12]
Jacky Keung	Researcher	[2]
Barbara Kitchenham	Researcher	[10]
Felicia Kurniawati	Researcher	[12]
Mark Staples	Practitioner	[10]
Liming Zhu	Researchers	[12]

5. RESULTS

While reading their chosen paper, each person in the group took one of the perspectives (self-chosen while ensuring both perspectives are covered). The allocation to paper and perspective is shown in Table 8. Everyone who took the practitioner viewpoint had worked for some time in industry.

Although each person reviewed their chosen ISESE paper from a particular perspective, in the inspection meetings (first the research perspective and next the practitioner perspective), they were encouraged to contribute to the discussion of the other perspective. We had originally planned for each person to provide a written list of issues/defects from their allocated perspective. This was done for the first two inspections but not done for the last 3 inspections. In practice, we worked through each of the questions, discussed any issues arising and agreed whether the question raised any problems or identified defects with the guidelines. After the first two inspections, we did not attempt to allocate individuals to specific perspectives.

The final inspection taking the author perspective proceeded differently. Again we used the ISESE papers to assist our understanding of the author perspective but instead of asking questions we discussed each section of the guidelines sequentially.

The perspective-based inspections using the Researcher, Practitioner, Meta-analyst, Replicator and Reviewer found 42 unique issues that we believed suggested the guidelines should be amended or clarified. Examples of these issues are shown in Table 9, and the full list is available from the first author. The Researcher perspective identified 13 possible amendments, the Practitioner / consultant perspective identified 21 possible amendments, the Meta-analyst perspective identified 4 possible amendments, the Replicator identified 3 possible amendments and the Review perspective identified 1 possible amendment. We also identified 8 items we classified as defects (see Table 10).

The inspection based on the author's perspective re-iterated many issues noted previously. In particular, we were concerned about suggestions to impose reporting structures that were incompatible with those used in other disciplines, such as the template structure for reporting research objectives and the section headings (see [4] and [8] for more conventional section headings). The problem of possible duplication was also reiterated.

The main issues that were not raised previously were that:

- The relationship between the "Experimental Design" and the "Execution" section needed to be clarified. If the first section was really the "Experimental Plan" and was fully reported, then the "Execution" section should be restricted to reporting deviations from the plan.
- The ordering of sections was not always appropriate, for example sometimes it is necessary to introduce the measurement concepts before specifying the hypotheses

Table 9 Examples of proposed amendments

ID	Amendment	Guidelines reference	Perspective (Question Number)
A2	The title needs to be informative. Specify the interventions (i.e. independent variables) and dependent variables [4] avoiding unnecessary redundancy.	n/a	Researcher (Q1)
A17	The scope information associated with the Related Work section (3.3) should request authors to comment on levels of industrial use the techniques being evaluated (including the control).	Related work (3.3)	Practitioner/Consultant (Q5)
A35	The information accompanying the guidelines should advice authors to report the number of experimental units.	Abstract (3.1) and Subjects 3.4.3	Meta analyst (Q1)
A40	There should be more advice on what is reported in Lessons learnt as opposed to other places for reporting deviations from plan.	Lesson learnt 3.7.3	Replicator (Q8)
A42	Authors should be advised to include all related work whether supportive or contradictory.	Motivation 3.3	Reviewer (Q3)

Table 10 Defects identified by Perspective-based Inspections

ID	Defect	Guidelines reference	Perspective (Question Number)
D1	The guidelines omit any reference to keywords.	None	Researcher (Q1)
D2	The guidelines do not conform to the classic reporting structure used by most other scientific domains (e.g. the IMRAD (Introduction, Material & Methods, Results, Discussion) format see [4[, [8]). Furthermore the deviation from the standard structure is not justified.	All	Researcher (Q3)
D3	The guidelines advice discussing validity and generalizability in five separate places. The guidelines introduce the possibility of considerable duplication and redundancy.	Context 3.2.3	Researcher (Q3)
D4	The guidelines do not address the needs of practitioner.	None	Practitioner (Q10[3])
D5	The guidelines do not require that the tasks the technology addresses are described.	Context 3.2.3	Practitioner (Q16)
D6	The guidelines do not require that the treatments (or level) be defined in operational terms.	Related work 3.3	Practitioner (Q18)
D7	Data set reduction is too specific since the authors needs to report procedures for data transformation, handling missing values etc. Change the heading for section 3.6.2 to Data set preparation.	Section 3.6.2	Meta-analyst (Q2)
D8	In several places the information accompanying the guidelines references the advice of other authors. However, it does not specify whether the advice should be followed or not (nor whether advice from different authors is contradictory). If the guidelines are recommending adoption of suggestions from other authors, they should say so explicitly.	None	Meta-analyst (Q2)

6. DISCUSSION AND CONCLUSIONS

The most significant defects are D2, D3, D4 and D8. D2 arises because the guidelines are inconsistent with reporting standards used by other experimental disciplines. It is a very significant step to disassociate our discipline with the standards used by all other scientific disciplines. We need to be sure that this step is necessary. At the least we need to articulate the reasons for this divergence, so software engineering researchers and practitioners understand why it is necessary. D3 is an important issue because it is an area that if not addressed may result in guidelines that make reporting experiments worse than it is currently. D4 is a general problem but a significant one. If we cannot write so that practitioners can understand and use our results, empirical software engineering is not very useful. D8 concerns the general principles of guidelines and standards – it should be clear what is mandated and what is optional.

Defects D5, D6 and D7 could easily have been classified as possible amendments. D5 and D6 are both related to the reporting of the technology or technologies being evaluated. If such technologies are not properly described it is difficult for practitioners to use them. D7 was a specific example of an issue that arose for several of the heading where the guidelines were too specific and should have used more general terms. Another example is the use of the term "subjects" rather than "experimental units". This raises another general issue that the guidelines are people/team centric. They do not address well the large number of technical tool "experiments" that get done in the "systems research" parts of the Software Engineering discipline (of which [10] is an example). Are these considered different types of studies? If so, it would be useful to clarify this in the scope of the guidelines; if not, the guidelines should be amended to make them more relevant to systems researchers.

Issues arising from the Author's perspective identified problems with potential duplication of information. Guidelines need to be very clear about what information goes into which section. This is a problem for the "Experimental Design" and "Execution" sections as well as the numerous validity sections.

Our results suggest that the main problems with the current version of the guidelines are:

1. Relationships among the individual elements are not clear. Thus, it is difficult to be sure what information to put in which section. There is also a risk that the guidelines will result in unnecessary duplication which would make experimental reports less readable.
2. In places, the guidelines require us to adopt reporting standards that are inconsistent with those of other disciplines. For example the suggested headings are inconsistent with the IMRAD standard (see [4] and [8]). We need to be absolutely certain that this is a good idea.

Our results suggest that the guidelines need to be revised. Any revised guidelines will need to be subjected to further theoretical and empirical validation if they are to be generally accepted. We also need to review research results in other disciplines that might provide additional justification for the guidelines structure and contents. For example, there have been numerous studies to assess the value of structured abstracts, summarized by Hartley [5].

[3] This defect was reported against Q10 but arose as a result of the accumulated problems addressing previous questions.

Our choice of evaluation method (inspection using different perspectives) seemed to work well for an initial theoretical validation. Our approach of multiple inspections fitted well with the training element of our evaluation exercise but is not an essential element of an inspection-based evaluation. It would be much quicker to perform a single inspection with individuals each taking a different perspective. We suggest that a similar inspection-based evaluation should be performed on the revised guidelines. Furthermore, the software engineering community should also undertake empirical validations of the proposed reporting guidelines, once there is a version that has been theoretically evaluated.

An important issue raised by the evaluation exercise is that of the Practitioner/Consultant viewpoint. The guidelines did not fit this perspective well. Attempts to address this perspective would make papers much longer and probably more complex. Would it be better to have different standards for practitioner-oriented papers? On the one hand, it can be argued that experiments in software engineering are not relevant to practitioners because they usually involve students, and/or simplified tasks and materials, and/or unrealistic settings. This would suggest practitioners only want to read case studies or industrial surveys. On the other hand, even if controlled experiments are not representative of industry practice, they provide proof of concept information without which industry is unlikely to undertake any realistic case studies. It may be that the only sensible course is to re-write research results for practitioner-oriented magazines (as long as copyright issues are addressed).

This paper has evaluated guidelines for controlled experiments. However, we believe that software engineering needs reporting guidelines for other types of empirical studies (in particular, case studies performed in industrial settings and industry surveys), not least because these types of studies are of most relevance to practitioners. We believe that many of the perspective-based questions related to researchers, practitioners, and reviewers are quite general (with the exception of questions that relate specifically to the methodology used for formal experiments) and can be used to help evaluate reporting guidelines developed for other forms of empirical study. Even the meta-analyst perspective and the replicator perspective are relevant to other forms of study although the questions would need to be revised. In particular, any attempt to construct and evaluate guidelines for industrial case studies and surveys should ensure that the practitioner perspective is fully considered.

7. ACKNOWLEDGEMENT

National ICT Australia is funded through the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council.

8. REFERENCES

[1] Zeiad Abdelnabi, Giovanni Cantone, Marcus Ciolkowski, Dieter Rombach. Comparing Code Reading Techniques Applied to Object-Oriented Software Frameworks with Regard to Effectiveness and Defect Detection Rate *Proceedings ISESE 04*, 2004.

[2] Silvia Abrahao, Geert Poels, and Oscar Pastor. Assessing the Reproducibility and Accuracy of Functional Size Measurement Methods through Experimentation, *Proceedings ISESE 04*, 2004.

[3] Tore Dybå, Vigdis By Kampenes, Dag I.K. Sjøberg. A systematic review of statistical power in software engineering experiments. *Information and Software Technology*, in press.

[4] Peter Harris. *Designing and Reporting Experiments in Psychology*. 2nd Edition, Open University Press, 2002.

[5] James Hartley. Current findings from research on structured abstracts. J. Med. Libr. Assoc. 92(3), July 2004, pp 368-371.

[6] Andreas Jedlitschka and Dietmar Pfahl. *Reporting Guidelines for Controlled Experiments in Software Engineering*. IESE-Report IESE-035.5/E, 2005.

[7] Barbara Kitchenham, Shari Lawrence Pfleeger, Lesley Pickard, Peter Jones, David Hoaglin, Khaled El Emam, and Jarrett Rosenberg. Preliminary Guidelines for Empirical Research in Software Engineering. *IEEE Transactions on Software Engineering*, 28(8) August 2002, pp721-734.

[8] David Moher, Kenneth F Schultz, Douglas Altman. The CONSORT statement: revised recommendations for improving the quality of reports of parallel-group randomized trials. *The LANCET*, 357, April 14, 2001, pp. 1191-1194.

[9] Pickard, L.M., Kitchenham, B.A., and Jones, P. (1998) Combining Empirical Results in Software Engineering. *Information and Software Technology*. Vol 40 No 14, pp 811-821.

[10] Patrick J. Schroeder, Pankaj Bolaki, and Vijayram Gopu. Comparing the Fault Detection Effectiveness of N-way and Random Test Suites, *Proceedings ISESE 04*, 2004.

[11] Dag I.K. Sjoberg, Jo E. Hannay Ove Hansen, Vigdis By Kampenes, Amela Karahasanovic, Nils-Kristian Liborg, Anette C. Rekdal. A Survey of Controlled Experiments in Software Engineering *IEEE Transactions on Software Engineering*, September 2005 (Vol. 31, No. 9) pp. 733-753.

[12] Jan Verelst. The Influence of the Level of Abstraction on the Evolvability of Conceptual Models of Information Systems. *Proceedings ISESE 04*, 2004.

[13] Wholin, C., Petersson, H., and Aurum, A. Combining data from reading experiments in Software Inspections. In Juristo, N. and Moreno, A. (eds.) *Lecture Notes on Empirical Software Engineering*, World Scientific Publishing, October 2003.

[14] Wholin, C., Runeson, P., Höst, M., Regnell, B., and Wesslén, A. Experimentation in Software Engineering. An Introduction. Kluwer Academic Publishers, 2000.