Understanding the impact of industrial context on software engineering research: some initial insights

Technical Report No.: 390

Austen Rainer
Department of Computer Science
University of Hertfordshire
Hatfield Campus
College Lane
Hatfield
AL10 9AB
U.K.

a.w.rainer@herts.ac.uk
+44 (0) 1707 284763

March 2003

Contextual variables appear to strongly impact the results of software engineering research. But little attention has been directed at better understanding this apparent impact. This report reviews the small body of research that has investigated industrial context. The report is something of a work in progress and, therefore, lacks some coherence. Therefore, a summary of insights is provided at the beginning of the report, with an appendix containing the remainder of the work.

Understanding the impact of industrial context on software engineering research: some initial insights

Austen Rainer

Contextual variables appear to strongly impact the results of software engineering research. But little attention has been directed at better understanding this apparent impact. This report reviews the small body of research that has investigated industrial context. The report is something of a work in progress and, therefore, lacks some coherence. Therefore, a summary of insights is provided at the beginning of the report, with an appendix containing the remainder of the work.

1. Summary of insights

This section summarises the material presented in Appendix A.

1. Recently, there have been recommendations from researchers to specify as much of the industrial context as possible. The most explicit recommendations have come from Kitchenham et al., [1] as part of a set of guidelines on empirical research in software engineering, and from Basili and Lanubile [2] in the Quality Improvement Paradigm. Kitchenham et al. state that researchers need to identify the particular factors that might affect the generality and utility of the findings of a study. There is an implication that the industrial context of a study 'contains' many of the factors that would affect generality and utility. According to Basili and Lanubile [2], the results of almost any process depend to a large degree on a potentially large number of context variables. As a result, one cannot a priori assume that the results of any study apply outside the specific environment in which that study was conducted.

2. There are a small number of specific definitions of context. For example, Briand et al. [3] define organizational context as the characteristics of relationships between process participants. Within this definition, examples of organizational context include management hierarchy, the structure of ad hoc working groups, and seating arrangements.

3. There are a small number of frameworks of industrial context. These consist of Kitchenham et al.'s [4] incomplete ontology of software maintenance, Basili et al.'s [2] framework for organising experimental studies of software reading techniques, and de Almeida et al.'s [5] ontology of the software process. (de Almeida et al.'s work is not discussed in the appendix. Their ontology was not explicitly presented as a framework of industrial context by its authors.)

4. No explicit method for 'discovering' industrial context has been developed within software engineering research. Nevertheless, there is a powerful method already available: the case study research strategy. Yin [6] states:

   > "… you would use the case study method because you deliberately wanted to cover contextual conditions – believing that they might be highly pertinent to your phenomenon of study." ([6], p. 13)

   There may also be other methods (e.g. archival analysis, document analysis, textual analysis) that could be used to conduct a systematic, formal literature review and identify contextual factors from previous literature in software engineering. Such methods need to be identified.

5. Industrial context relates to experimental studies and observational studies in different ways. For observational studies, the researcher has some access to the actual context (cf. the use of the case study research strategy). The nature of experimental studies is such that context is deliberately ‘removed’ from the study. Therefore, elements of a possible context are either modelled explicitly in the experimental design (probably as confounding variables) or the researcher must take account of possible industrial contexts when interpreting the validity and utility of the results of the experimental study.

6. Industrial context seems to relate closely with issues of validity, particularly the validity of subjects in experimental studies (e.g. students vs. professional practitioners) and the validity of the experimental situation (e.g. that the activities within the experiment are representative of real software development, and that the subjects are appropriately motivated to do the work in a professional way). Industrial context may also relate to internal validity of experimental studies, as the context may confound the relationship between independent and dependent variables.

7. Some researchers [7] are calling for the collection of a standard set of measures for empirical studies. One implication is that this standard set should capture at least some of the industrial context. Also, some of these measures are, presumably, intended to be descriptive (i.e. the measures describe aspects of the empirical study or its setting) and are not necessarily intended to be used immediately for analysis. It may be, for example, that once several studies have collected a standard set of measures some of these measures can be used in a subsequent meta-analysis.

8. A brief review of the literature suggests that researchers typically use the term ‘industrial context’ when they are referring to:

    1. Phenomenon that are not the focus of an empirical investigation, and yet are considered to have some relevance to the focus of the investigation, where:
    2. The phenomenon is not well-understood, and
    3. The connections between the phenomenon and the focus of the study are not well-understood, because
    4. Resource constraints (e.g. effort, availability of skilled resource) and psychological constraints (e.g. bounded rationality) mean that the researcher must often direct their efforts almost entirely at the focus of the study and cannot consider the context of that study.

    Things that are classified as context are things that seem to lie on the ‘edge’ of our intellectual focus. (This implies two stages to an enquiry: attending to the focus of the enquiry and then attending to the *boundaries* of that focus.)

9. The context of a study seems to relate to the main units of analysis, which in turn depend on the level of enquiry and the main research questions [6]. Where researchers study the same phenomenon, they frequently study it from different perspectives and at different levels of enquiry. For example, Seaman and Basili write:

    > “In software engineering studies of process, units of analysis range over all levels, from entire projects to enactments of a single subprocess.” ([8], p. 199)

10. A number of researchers have analysed and discussed the lack of evaluation that is conducted in empirical studies of software engineering e.g. [9-12]. Explicitly modelling context would provide one foundation for evaluating the study: researchers can then use that model to evaluate the generalisability and utility of their findings. Indeed specifying a model of contextual 'variables' can help to explicitly consider threats to validity. Modelling context may also make it easier for researchers to connect different studies e.g. through a mutually relevant context ‘variable’, or perhaps because a study variable in one study is a contextual variable in another.

11. Some preliminary guidelines are also suggested, based on:

    1. Fenton and Pfleeger's [13] generic model of process.
    2. Kitchenham et al. [1] preliminary guidelines for software engineering research.
    3. Kitchenham et al. [4] use of 'competency questions' to develop their partial ontology.
    4. Robson's [14] identification of four broad threats to external validity.
    5. Yin's [6] discussion of the case study research strategy, in particular his discussion of the unit of analysis.
    6. Miles and Huberman [15] recommendations the use of context charts to visualise the context of a study.

Appendix A

1 Introduction

Recently, there have been recommendations from researchers to specify as much of the industrial context of an empirical study as possible. The most explicit recommendations have come from Kitchenham et al. [1] as part of a set of guidelines on empirical research in software engineering, and from Basili et al. in his Quality Improvement Paradigm e.g. [16]. Kitchenham et al. state that researchers need to identify the particular factors that might affect the generality and utility of the findings of a study. There is an implication that the industrial context of a study 'contains' many of the factors that would affect generality and utility.

A cursory review of the software engineering literature indicates that many studies, particularly field studies, refer to the industrial context of the study. (Of course many of the studies do not use the literal term 'industrial context'.) Given the frequent reference to industrial context and the recommendation to specify as much of the industrial context as possible, it is surprising that there is very little work directed at defining the concept of context, developing a standard framework of context that can be applied to many studies, or developing a method for identifying and specifying the context for a particular study. In this paper, I direct attention at these three issues.

2 Two existing frameworks on industrial context

Broadly, there are two existing framework of industrial context in the software engineering literature: an incomplete ontology of software maintenance, and a partial problem space addressed by software reading techniques.

2.1 A partial ontology of software maintenance

Kitchenham et al. suggest that empirical studies of maintenance are difficult to understand unless the contexts of the studies are fully defined. For this reason, they chose to develop an ontology of maintenance, for the purpose of identifying contextual factors that influence the results of empirical studies of maintenance. Kitchenham et al. believe that the ontology provides a number of benefits for the research community, two of which are particularly relevant here. First, the ontology allows researchers to provide a context within which specific questions about maintenance can be investigated. Second, the ontology provides a standard framework to assist the reporting of empirical studies. In developing their ontology, Kitchenham et al. draw heavily from the work of de Almeida, de Menezes and da Rocha [5], who have previously developed a software process ontology.

Kitchenham et al. recognise that their ontology is incomplete. Kitchenham et al. state that they have not attempted to validate the ontology, recognise that there may be other factors that they have not considered, and emphasise that is it the responsibility of individual researchers to attempt to identify any special conditions that apply to his/her results.

2.2 A framework for organising experimental studies of software reading techniques

According to Basili and Lanubile [2], the results of almost any process depend to a large degree on a potentially large number of context variables. As a result, one cannot a priori assume that the results of any study apply outside the specific environment in which that study was conducted. Basili and Lanubile go on to propose a number of conditions that must be met in order for the research community to build up comprehensive bodies of knowledge in areas of software engineering. Two conditions are particularly relevant here. First, Basili and Lanubile propose that detailed hypotheses are written in a context that allow for a well defined experiment. Second, Basili and Lanubile propose the use of context variables that can be changed to allow for variation of the experimental design, that can counter threats to validity, and that can reflect the specifics of the process context.

Basili et al. developed a tree-structured framework that described the problem space with which they intended to deal. The framework has evolved as new experiments have been attempted. For Basili and Lanubile, the tree structure provides a number of advantages:

- More and more detail is specified as one moves down the tree.
- The framework allows one to define classes of experiments and combine them back to high-level hypotheses.
- The tree structure helps keep the relevant context variables organized and thus helps in assessing the contribution of a particular experiment.
- The tree structure serves as a basis for comparing experiments.
- The tree structure summarises the amount of experimentation in an area and illustrates the degree of 'coverage' of that area.

Basili and Lanubile also discuss the importance of the motivation of subjects in experiments. Ideally, the study of a software engineering process should assess that process (or some related aspect of the process, such as a technology) within the context of real work practices. This means that experimenters need subjects to perform at a level that is representative of their professional work. Ideally, the experimental setting should either reflect the organisational setting of should allow the subjects (as professional practitioners) to recognise some professional benefit from doing the experimental activities. This would motivate them to put a realistic amount of effort and thought into the activities.

Basili et al. (e.g. [16]) also recognise the importance of context in their six-step Quality Improvement Paradigm (QIP), the first step being *Characterize the environment*. This paradigm has been used within the context of another of Basili et al.'s 'technologies' i.e. the experience factory.

2.3 A comparison of the two frameworks

Both Kitchenham et al. and Basili et al. recognise the importance of context to empirical studies. Given this common recognition, it is very interesting to see the development of two very different frameworks. The two frameworks have very different structures, have different scopes (the Kitchenham et al. framework has a broad scope focusing on maintenance, the Basili et al. framework has a specific scope, focusing on software reading) but also have different resolutions (the Kitchenham et al. framework considers a broader range of things within software maintenance, whilst the Basili et al. framework concentrates almost exclusively on technical aspects). The Kitchenham et al. framework has, in principle, a wider applicability and yet the Basili et al. framework is the one most easily connected with particular empirical studies. The Kitchenham et al. framework also seeks to define the impact of factors on software maintenance, whereas the Basili et al. framework appears to be a taxonomy. The Basili et al. framework is more obviously tied to a specific kind of study i.e. experimental study. Basili et al. has also provided some interesting insights into *subjects* and their role in the external validity of empirical studies.

3 Other work on industrial context

Several researchers have directly or indirectly sought to define context, and to explain the importance of context to empirical studies.

3.1 Briand et al.'s definition of organizational context

Briand et al. [3] define organizational context as the characteristics of relationships between process participants. Examples of organizational context include management hierarchy, the structure of ad hoc working groups, and seating arrangements. Briand et al. ([3], p. 133) state:

> "It is not possible to fully understand and analyze such process issues as information flow, division of work, and coordination without including the organizational context in the analysis." ([3], p. 133)

Briand et al. ([3], p. 133) add that the organizational context in which a development process executes has received less attention in the literature.

Briand et al.'s definition focuses on characteristics of relationships between people that work together within one process. In some respects this is a narrow definition of context: for example, the definition doesn't obviously refer to project characteristics (e.g. in-house development, software supplier), product characteristics (such as the type of product being developed), or the industry within which the product will be used. But it's also a narrow definition in that it is not clear about the kinds of relationships that should be considered. For example, there may be social relationships between people and possibly political relationships and these may influence the formation of ad hoc working groups. The point I want to make here is not that Briand et al. have provided an incomplete definition but rather that a complete definition of context is particularly difficult to develop. I will return to this point later in the paper.

3.2 Curtis et al.'s behavioural model

Curtis et al. [17] proposed a five layer behavioural model of the software process, proceeding from the cognitive and affective behaviour within an individual, to the behaviour of teams, projects, organisations and then the business milieu. Curtis et al. stated that the behavioural model would lie orthogonal to the technical process of developing the software artefacts. For each level of the behavioural model, the higher levels can be considered to provide a context for the lower levels. Similarly, each level of the behavioural model can provide some context for studies that focus on the technical development of software artefacts.

3.3 Seaman and Basili's definition of organizational context

Seaman and Basili [8] state that one often overlooked component of both development and maintenance processes is the organisational context in which they are enacted. Similar to Briand et al., Seaman and Basili define organisational context (Seaman and Basili also use the term 'organisational structure') as follows:

> "… an efficient flow of information [between people involved in the process] is crucial, and this flow can be hindered or facilitated by the organizational relationships between maintainers and other parts of the organization. It is the network of these relationships (reporting relationships, physical proximity, past working relationships, etc.) that constitute organizational structure." ([8], p. 197-201)

(The similar definitions between Seaman and Basili, and Briand et al. are not surprising when one considers the co-authors involved in the two papers.)

3.4 Pickard et al.'s standard set of measures

Pickard et al. [7] consider approaches for combining results in empirical studies of software engineering. One critical issue identified by Pickard et al. is the need to agree a set of standard measures which are recorded for all empirical studies. Such a set of standard measures will eventually allow researchers to define appropriate sub-populations. Pickard et al. suggest that such a standard set of measures would be analogous to gender and age in medical studies. Pickard et al. write:

> "Although we can apply standard scientific techniques such as formally designed experiments or statistical data analysis techniques, it is not clear that our results have the same generality as empirical results in other disciplines. The problem arises from the difficulty of defining the population of software engineering subject and objects to which any results can properly be said to generalize." ([7], p. 811)

4 Problems with defining context

The lack of a definition of industrial context in the software engineering literature suggests that there is great difficulty in defining the term 'industrial context'.

4.1 The pragmatics of the term 'industrial context'

One place to begin is to consider the pragmatics of the term i.e. to consider when and where the term 'industrial context' (or similar terms) is used. A brief review of the literature suggests that researchers typically use the term 'industrial context' when they are referring to:

1. Phenomenon that are not the focus of an empirical investigation, and yet are considered to have some relevance to the focus of the investigation, where:
2. The phenomenon is not well-understood, and
3. The connections between the phenomenon and the focus of the study are not well-understood, because
4. Resource constraints (e.g. effort, availability of skilled resource) and psychological constraints (e.g. bounded rationality) mean that the researcher must often direct their efforts almost entirely at the focus of the study and cannot consider the context of that study.

Any investigation necessarily excludes things from that investigation. Use of the term 'industrial context' seems to be an implicit recognition that certain things have been excluded and that these things probably have some relevance to the study. Phrased another way, things that are classified as context are things that lie on the ‘edge’ of our intellectual focus. (This implies two stages to the enquiry: attending to the focus of the enquiry and then attending to the *boundaries* of that focus.)

When researchers refer to context, they seem to be referring to the (in Kitchenham et al.’s words) “... immense variety to be found in development procedures, organizational culture, and products.” (p. 723)

4.2 Units of analysis and their context

Yin [6] provides an extensive treatment of the case study research strategy. He writes that the definition of the main and embedded units of analysis of a case study, as well as the definition of the contextual events surrounding these units, depends on the level of enquiry. Furthermore, the main unit of analysis is likely to be at the level being addressed by the main study questions. Yin’s statements provide further insights into why the definition of context is difficult: the context of a study seems to relate to the main units of analysis, which in turn depend on the level of enquiry and the main research questions. Where researchers study the same phenomenon, they frequently study it from different perspectives and at different levels of enquiry. Indeed, formal, replicated studies are likely to be one of the few occasions where two separate studies approach the same phenomenon from the same perspective and level of enquiry. Therefore, it is with replicated studies that one is most likely to find similar contexts.

For example, Seaman and Basili write:

> “In software engineering studies of process, units of analysis range over all levels, from entire projects to enactments of a single subprocess.” ([8], p. 199)

4.3 Context as confounding factors

Kitchenham et al. [1] argue that the identification of standards for reporting research context requires further research in two areas: the identification of general confounding factors, and the specification of a taxonomy or an ontology of context. Fenton and Ohlsson [18] propose some hypotheses and provide evidence from two case studies to support their hypotheses. Fenton and Ohlsson recognise that they have not collected evidence on two confounding factors: test effort and operational usage. Fenton and Ohlsson’s study is relevant here for two reasons. First, the authors are working with context (as loosely defined here, and also by Kitchenham et al.) although not explicitly presenting their confounding factors as contextual. Second, they are being explicit about these factors, and this explicitness is a counter-example to the pragmatics of context that I discussed above. In other words, some studies are considering the problems of validity and utility, but not explicitly using terms such as ‘industrial context’ in their considerations.

Wohlin and Andrews [19] provide another example of the way that context is modelled. For each of 46 projects from one organisation, six success factors and 27 project characteristics were measured. Interestingly, of the 27 characteristics that Wohlin and Andrews investigated, they only identified two characteristics (“project planning” and “ability to follow the plans”) that affected project success. Clearly this is a small number of characteristics, and seems to contradict (superficially perhaps) Kitchenham et al.’s and Basili et al.’s claims about a potentially large number of contextual variables.

As another example, Deephouse et al. [20] investigated the relationship between software process and project performance taking account of the confounding factors of rework and project characteristics. Again, they collected information on a number of characteristics, and did not directly refer to context.

5 Defining context in terms of validity

In their preliminary guidelines, Kitchenham et al. state that researchers need to identify the particular factors that might affect the generality and utility of the findings of a study. Pickard et al. make a similar point:

> “The problem arises from the difficulty of defining the population of software engineering subject and objects to which any results can properly be said to generalize.” ([7], p. 811)

Context seems to be strongly related to the issue of being able to apply the results of a study to other situations, particularly other *real-world* situations. A specification of context should allow one to identify and evaluate threats to the validity of a study, particularly the external validity and ecological validity of that study.

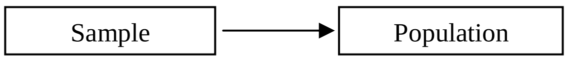

Figure 1 A visualisation of external validity

External validity, as used here, refers to the degree to which findings can be generalized from the specific sample in the study to some target population ([14], p. 46). A common criticism of using student subjects in a study is that they do not necessarily represent the population of professional practitioners. Porter and Votta [21] replicated previous studies but chose to use professional practitioners in their replication. (Basili’s comments on subjects’ motivation is relevant here.)

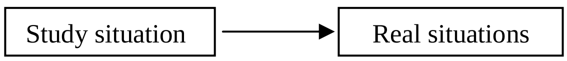

Figure 2 A visualisation of ecological validity

Ecological validity, as used here, refers to the degree to which the situation occurring in the study is representative of the real-world. Porter and Votta [21] attempted to ensure that the review process in their experiment was similar to the review process used in industry, by modelling the experiment’s inspection process on one used in several development organisations within AT&T.

More generally, the software process used in the study should be as close to the process used in industry. But this recommendation can conflict with the general strategy of experiments i.e. to remove confounding factors, leaving a ‘pure’ relationship between independent and dependent variables.

Table 1 Four threats to external validity

Threat	Description
Selection	The findings being specific to the group studied.
Setting	The findings being specific to, or dependent on, *the particular context* in which the study took place.
History	The specific and unique historical experiences may determine or affect the findings.
Construct effects	The particular constructs studied may be specific to the group studied.

Robson [14] suggests four threats to external validity (after LeCompte and Goetz, 1982). These threats are summarised in Table 1. Of these four threats, the second and third clearly relate to ecological validity and hence to context. The other two threats relate more to external validity, but this is also relevant to context.

6 Defining context in terms of experimental studies

According to Yin, "An experiment... deliberately divorces a phenomenon from its context, so that attention can be focused on only a few variables (typically, the context is 'controlled' by the laboratory environment)." ([6], p. 13)

In an experiment context is either deliberately removed or is deliberately modelled as one or more variables within the experimental model (and typically one or more confounding variables). When the experimenter comes to interpret the findings of the experiment, the experimenter will also then attend to contextual information that was not included in the experimental model.

So in one respect context is not relevant to experiments, because the experimenter is deliberately trying to rule out things (such as context) over which the experimenter has no control. But in another respect context is relevant to experiments, because the experimenter may try to include aspects of context as experimental variables, but also because the experimenter will eventually evaluate the results of the experiment into terms of real-world context.

6.1 An operational definition

Consider an experimental model of independent and dependent variables. Such a model would constitute the focus of an experimental study. In an experimental study using that model it is the researchers who, of course, control the variation in the independent variables. In a non-experimental situation, however, the researchers would not control the variation in the independent variables. In such a situation, the variation in independent variables would be caused by (or accounted for by) some other phenomenon. Another experimental study would model these other phenomenon could be modelled as independent variables. The independent variables of the original study would become dependent variables in the new study.

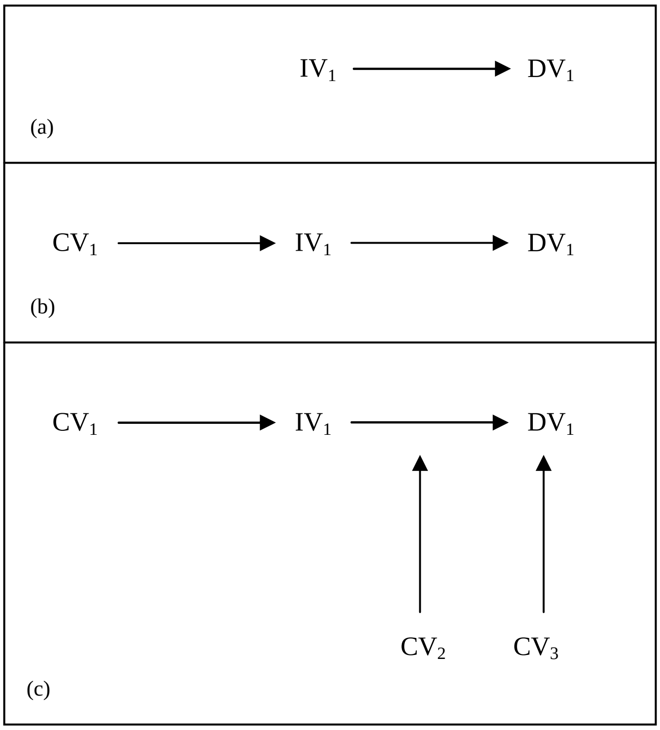

Figure 3 Experimental models

Fig. (a) shows a simple relationship between an independent variable (IV_1) and a dependent variable (DV_1). The experimenter would vary the values of the independent variable and observe the subsequent variation in the dependent variable.
Fig. (b) includes a contextual variable (CV_1). In reality, the values of the independent variable would not be manipulated by the experimenter but would be affected by other variables e.g. the contextual variable.
Fig. (c) indicates how contextual variables could affect other variables in the experimental model.

Figure 3 illustrates the relationship of independent, dependent, and contextual variables.

If one were to model industrial context within an experimental model, one would need to operationalise the context in terms of one or more variables, and then determine whether each of these contextual variables would be treated as an independent or dependent variable. More specifically, for each contextual variable, one would consider:

1. Whether the contextual variable had a *direct* (causal) effect on an independent variable.
2. Whether the contextual variable had a *direct* (causal) effect on a dependent variable.
3. Whether the contextual variable *directly* mediates the effect of an independent variable on a dependent variable.

The preceding discussion indicates that industrial context would be operationalised in terms of specific variables and these variables would be added to the experimental model (see Figure 3(c)). For an experimental model, there would be the requirement for the contextual variables to have a hypothesised causal effect on other variables in the model. We may wish to relax this requirement, and simply assert that the contextual variables are hypothesised to have some connection with the independent and dependent variables.

There needs to be some limit placed on the number of contextual variables included in an experimental model, otherwise the model could become unnecessarily complicated. The model could also become 'unbalanced' with an undue emphasis on contextual variables rather than experimental variables. For this reason, I have recognised that a contextual variable should, for example, have a direct causal effect on another variable in the model.

(Seaman and Basili, Briand et al. and Basili’s QIP are all taking an experimental approach to context.)

7 Two methods for specifying industrial context

Yin is very clear in his definition of a case study:

> "A case study is an empirical enquiry that:
> - investigates a contemporary phenomenon within its real-life context, especially when
> - the boundaries between phenomenon and context are not clearly evident." (p. 13)

In discussing industrial context and its relation to formal experiments, Kitchenham et al. write:

> "Unlike techniques developed in academia, industrial techniques have usually been developed in a rich complex setting. We need to understand how the technique works in an industrial setting *before* we develop an abstract version for formal experimentation." ([1], p. 723; emphasis added)

Put simply, I propose that the case study research strategy is one excellent method for specifying industrial context. It may seem rather flippant of me to suggest the case study research strategy. But the case study research strategy is very well-developed and powerful method and, as indicated by Yin, it is ideal for the problem in hand. Far better to use an existing research strategy, where that strategy is appropriate, than to attempt to develop an entirely new strategy.

One or more case studies could be used to help identify, clarify and define the boundaries between the phenomenon and its context. The case studies could also help identify, clarify and define possible confounding factors[1]. The results of the case studies can then be used to help design formal experiments, and subsequently evaluate the applicability and utility of those experiments' findings to real-world situations. Indeed Yin adds:

> "… you would use the case study method because you *deliberately* wanted to cover contextual conditions – believing that they might be highly pertinent to your phenomenon of study." (p. 13; emphasis added)

This is one area in which previous research can be particularly helpful, as case studies may have already been conducted. Even in the absence of case studies, previous studies may have sufficiently separated context and phenomenon, and identified and defined confounding variables, to make formal experiments appropriate.

As a second method for specifying industrial context, I would advocate the use of archival analysis (and related techniques) but apply those techniques to the existing research literature. Phrased another way, I suggest a more formal approach to literature review. Clearly, the relevance of this second method requires more thought.

8 Modelling context

8.1 The value of modelling context

A number of researchers have analysed and discussed the lack of evaluation that is conducted in empirical studies of software engineering e.g. [9-12]. Explicitly modelling context (either through applying a standard taxonomy or ontology, or through generating a model specifically for the study) would provide one foundation for evaluating the study: researchers can then use that model to evaluate the generalisability and utility of their findings. Indeed specifying a model of contextual 'variables' can help researchers (and, indeed, reviewers and readers) to explicitly consider threats to the validity of the respective study.

Modelling context may make it easier for researchers to connect different studies e.g. through a mutually relevant context 'variable', or perhaps because a study variable in one study is a contextual variable in another.

8.2 Two examples of a preliminary modelling notation

---

[1] Note two things here, relevant to context. Researchers refer to 'context' when the boundaries between phenomenon and real-world context are not clear; and when confounding factors are not known.

I have begun to sketch out some modelling notation for modelling context. I present two examples here:

1. Modelling the context of Fenton and Ohlsson's [18] study of faults and failures in a complex software system.
2. Porter and Votta's [21] replication of software requirements inspections.

8.2.1 Fenton and Ohlsson's study of faults and failures

Figure 4 A model of variables in Fenton and Ohlsson's [18] study

Figure 4 presents a simple model of the variables used in Fenton and Ohlsson's [18] study.

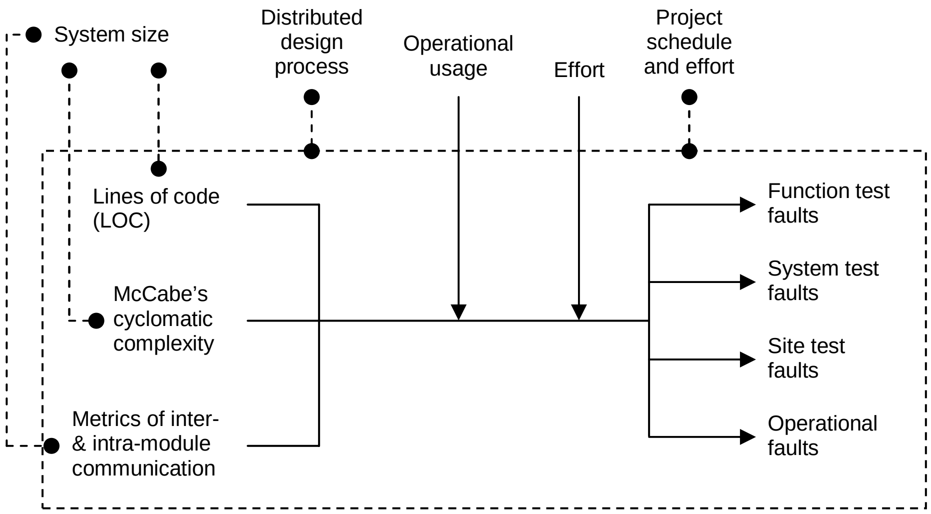

Figure 5 Model of variables, with context

The model presented in Figure 5 is developed from the information presented on p. 799 and p. 800 of Fenton and Ohlsson's [18] paper. In this model, I am trying to include 'contextual variables'.

- I have suggested that the size of the system may connect with measures of size and complexity, but not suggested the substance of the relationship e.g. whether the relationship is causal or predictive or merely associative.
- Fenton and Ohlsson have suggested that operational usage and effort are explanatory variables. I have modelled those here, to some degree.
- Fenton and Ohlsson also briefly described some characteristics of the process, and the project. Again, I have modelled those here.

8.2.2 Porter and Votta's replication of software requirements inspections

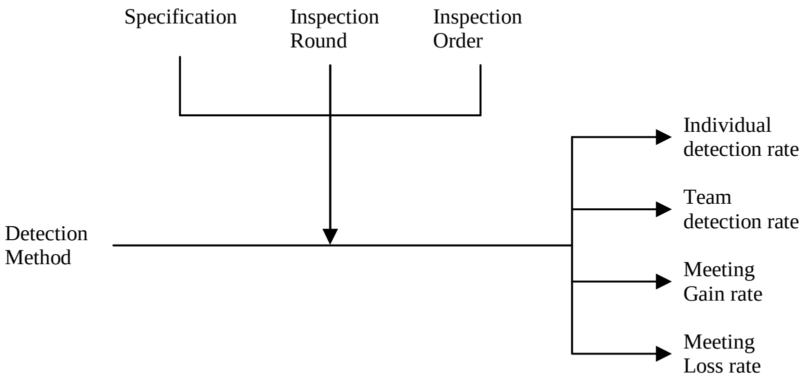

Figure 6 Model of variables in Porter and Votta's study

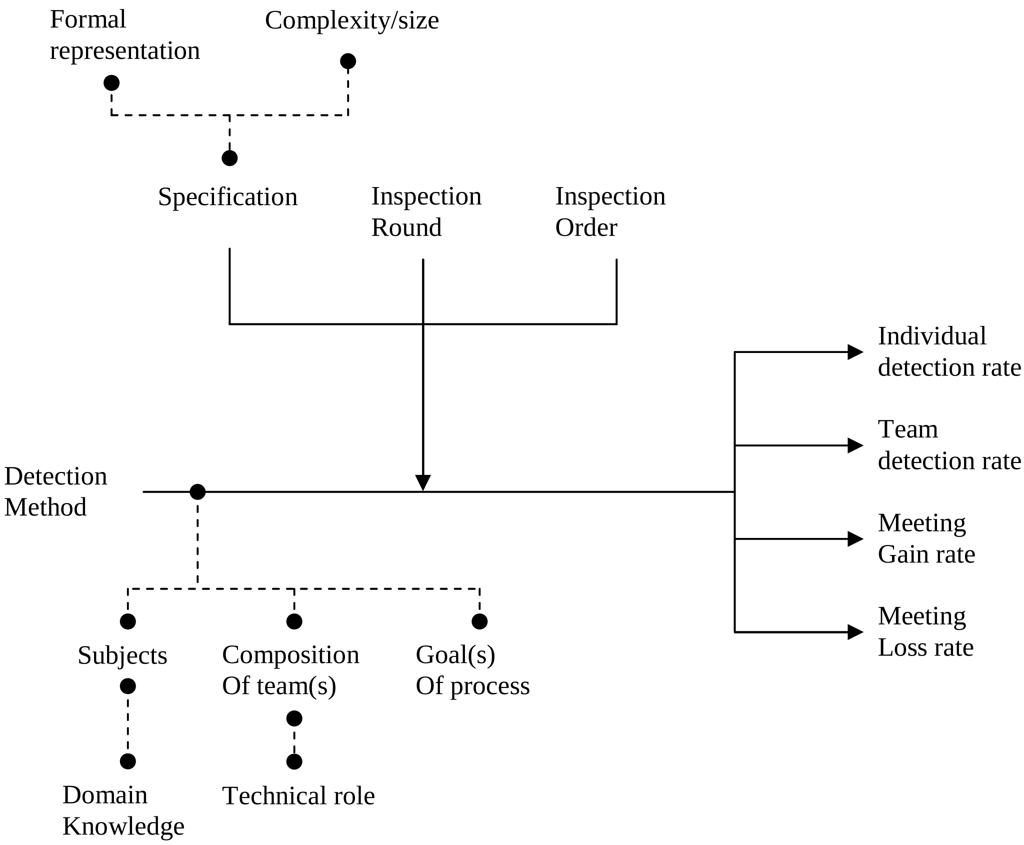

Figure 7 Model of Porter and Votta [21], including contextual variables

In this model, I am trying to add 'contextual variables' to the original experimental model. These variables are identified from p. 361 of Porter and Votta's [21] paper. (An interesting aside is that the paper does not talk about context explicitly – instead it discusses threats to external validity[2]. I note that Kitchenham et al.'s point about specifying context seems to be about considering the generalisability and utility of findings which, in part at least, relates to external validity.)

Note that I am representing *connections* rather than causal relationships.

[2] Would this be something worth talking about in the guidelines, and also in the archival analysis method?

9 Guidelines for defining industrial context

From the preceding review of literature, a number of tentative guidelines can be suggested:

1. Where a *process* is being investigated, Fenton and Pfleeger's [13] generic model of process may be useful. The model comprises five components: inputs to the process, outputs from the process, resource used by the process, controls on the process, and the process itself. These five components suggest five general areas for which researchers can consider collecting information.

2. In their preliminary guidelines on software engineering research, Kitchenham et al. [1] provide several examples of contextual factors. They are:
   - The industry within which the products are used (or will be used).
   - The nature of the software development organisation.
   - The skills and experience of software practitioners.
   - The type of software products used.
   - The software processes being used.

   Kitchenham et al. also recommend that the experimental unit is explicitly defined to help one reduce the chance of mistaking the unit of analysis. And that the experimental unit is clearly separated from the unit of observation. As an example of the mistakes than can arise, Kitchenham et al. describe how survey researchers can confuse the unit of observation (the individual) with the unit of analysis (the organisation). Being clear about the experimental and the observational unit can help clarify the context.

3. During the development of their partial ontology, Kitchenham et al. [4] used three 'competency questions', which again might be useful (they are paraphrased here to make them more generic to software development):

   - Would variations in this factor influence empirical studies of software development or software maintenance? Would variations in this factor influence productivity, quality, or efficiency?
   - What is the nature of the variations in this factor?

4. Robson [14] identified four broad threats to external validity. One can use these to identify particular contextual factors. See section 5, Table 1 for more information.

5. Yin [6] discusses the case study research strategy in detail and devotes attention to defining the unit of analysis. His advice on defining units of analysis are also helpful in defining the context.

   - As a general guide, the definition of a unit of analysis (and therefore of the case) is related to the way the initial research questions have been defined.
   - The more a study contains specific propositions (specific hypotheses) the more the study will remain within feasible limits and the easier it should be to identify contextual variables relating to those propositions.
   - For almost any topic that might be chosen, specific time boundaries are needed to define the beginning and the end of the case.
   - Most researchers will want to compare their findings with previous research. Therefore, each case study and unit of analysis either should be similar to those previously studied or should deviate in clear, operationally defined ways. In this way, previous literature can become a guide for defining the case and the unit of analysis.

6. Building a model of the context can help to clarify the relationships of the contextual variables with the study variables. For example, Miles and Huberman [15] recommend the use of context charts to visualise the context of a study.

10 Further research

Clearly, there is considerable scope for further research in this area.

- Further thought directed at the types of connections between contextual variables and study variables. In an experimental model, the researcher would be looking to model a causal, predictive or at least associative relationship. But there may be more ‘relaxed’ relationships worth modelling.
- A modelling notation for modelling contextual ‘variables’ and, in the case of experimental studies, explicitly connecting those contextual variables to the study variables. Note that contextual variables here would refer to those variables that are *not* included in the experimental model.
- As noted by Kitchenham et al. there is also a need to start to specify measurements for the contextual variables.
- By the authors’ own recognition, the Kitchenham et al. framework has not been validated, so there is a need to validate the framework.
- Similarly, while Basili et al.’s framework is being used to help design experiments and organise results, an independent validation of the framework may also be valuable.

References

1. Kitchenham, B., et al., *Preliminary guidelines for empirical research in software engineering.* IEEE Transactions on Software Engineering, 2002. **28**(8): p. 721-734.
2. Basili, V.R., F. Shull, and F. Lanubile, *Building Knowledge through Families of Experiments.* IEEE Transactions on Software Engineering, 1999. **25**(4): p. 456-473.
3. Briand, L., et al. *Characterizing and assessing a large-scale software maintenance organization*. in *17th International Conference on Software Engineering (ICSE-17)*. 1995. APril 23-30, 1995, Seattle, Canada.
4. Kitchenham, B., et al., *Toward an ontology of software maintenance.* Journal of Software Maintenance: Research and Practice, 1999. **11**: p. 365-389.
5. de Almeida Falbo, R., C.S. de Menezes, and A.R.C. da Rocha. *Using ontologies to improve knowledge integration in software engineering environments*. in *4th International Conference on Information Analysis and Synthesis.* 1998. Caracus, Venezuela: International Institute of Informatics and Systemics.
6. Yin, R.K., *Case Study Research: Design and Methods*. 2nd edition ed. 1994: SAGE Publications.
7. Pickard, L., B. Kitchenham, and P. Jones, *Combining empirical results in software engineering.* Information and Software Technology, 1998. **40**(14): p. 811-821.
8. Seaman, C. and V.R. Basili, *The study of software maintenance organizations and processes.* Empirical Software Engineering, 1997. **2**(2): p. 197-.
9. Zelkowitz, M.V. and D. Wallace. *Experimental validation in software engineering*. in *Empirical Assessment in Software Engineering (EASE-97)*. 1997. University of Keele.
10. Tichy, W.F., et al., *Experimental Evaluation in Computer Science: A Quantitative Study.* Journal of Systems Software., 1995. **28**(1): p. 9-18.
11. Glass, R.L., *A Structure-based critique of contemporary computing research.* Journal of Software Systems, 1995. **28**(1): p. 3-7.
12. Fenton, N., S.L. Pfleeger, and R.L. Glass, *Science and substance: A challenge to software engineers.* IEEE Software, 1994. **11**(4): p. 86-95.
13. Fenton, N. and S.L. Pfleeger, *Software Metrics: A Rigorous and Practical Approach*. 2 (revised printing) ed. 1997, London: PWS Publishing Company.
14. Robson, C., *Real World Research: A Resource For Social Scientists and Practitioner-Researchers*. 1993, Oxford UK: Blackwell.
15. Miles, M.B. and A.M. Huberman, *Qualitative Data Analysis*. 2nd ed. 1994, London: SAGE Publications.
16. Basili, V.R. and G. Caldiera, *The Experience Factory Strategy and Practice*. 1995, University of Maryland. p. 41.
17. Curtis, B., H. Krasner, and N. Iscoe, *A field study of the software design process for large systems.* Communications of the ACM, 1988. **31**(11): p. 1268-1287.
18. Fenton, N. and N. Ohlsson, *Quantitative Analysis of Faults and Failures in a Complex Software System.* IEEE Transactions on Software Engineering, 2000. **26**(8): p. 797 - 814.
19. Wohlin, C. and A.A. Andrews, *Prioritizing and assessing software process success factors and project characteristics using subjective data.* Empirical Software Engineering, 2003. **8**: p. 285-308.
20. Deephouse, C., et al., *Software processes and project performance.* Journal of Management Information Systems, 1996. **12**(3): p. 187-205.
21. Porter, A.A. and J.G. Votta Jr., *Comparing detection methods for software requirements inspections: a replication using professional subjects.* Empirical Software Engineering, 1998. **3**: p. 355-379.