A Model of Software Process Improvement from a Systemic Perspective

Peter D.C. Bennetts[1]
A. Trevor Wood-Harper[2]

[1]Gloucestershire Business School
Cheltenham and Gloucester College of Higher Education
Cheltenham, United Kingdom.
Email: pbennetts@chelt.ac.uk

[2]School of Accounting and Information Systems
University of South Australia, Adelaide, Australia
Email: trevor.wood-harper@unisa.edu.au
and Information Systems Institute
University of Salford, Salford, United Kingdom
Email: a.t.wood-harper@salford.ac.uk.

Abstract

This paper develops a model of the processes of information systems development through the use of Soft Systems Methodology and the Goals / Questions / Metrics Paradigm. It is recognised that each organisation is different and pursuing different goals, so the model needs to be contingent. Further, the model supports an on-going programme of enhancing software quality. The model is called the Contingent Factors Model.

Currently, this model has not been used in a commercial setting. This paper argues that any such intervention should be performed using action research.

Keywords

Quality, IS Development Methodology, Soft Systems, Multiple Perspectives

INTRODUCTION AND BACKGROUND

Looking back at the development of computing, it seems clear that it has always been difficult to develop new systems. In the late 1960s and early 1970s, the DoD were so concerned about the perceived failure of developers to create systems to budget, time and functionality, that they convened a NATO working conference whose title introduced the concept of a "software engineer" (Naur et al. 1976). In support of this "re-launch" of information systems development (ISD), many argued that more training was required, together with more mathematics and formality (Dijkstra 1976, Hoare 1982). While improvements have accrued, many systems are still considered to be failures, and for reasons that are not addressed by a purely technical perspective (Clegg et al. 1996, Standish Group 1995). The authors take the view that in order to improve new systems' failure rate, both technical and non-technical issues need to be explicitly addressed. This leads to a recognition of systems approaches as being pertinent to support the development of ISD. In particular, Checkland and Scholes' (1990) Soft Systems Methodology (SSM), is appropriate.

Lyytinen (1988) recognised that SSM gave "a better and more complete identification of stakeholders, a deeper insight into the dynamic nature of IS problems and a careful perception of several ... problems which are often ... unnoticed". For SSM, a particular problem situation (in this case ISD) is examined through an analysis of the perceived corresponding Human Activity System (HAS), which represents both product and process issues. Software Quality Assurance is seen as an element of the monitoring and control processes of a HAS for ISD. As well as offering a means to resolve problems, SSM can be seen as a means of describing an appropriate set of activities that should be taking place in ISD i.e. as a metaphor for ISD. The aim is to develop as wide an approach to software quality as possible, through the use of success factors, which is sensitive to the characterisation of software quality developed by an organisation and its development personnel.

It is accepted that there is a tendency for software engineers to try to get others to address the non-technical issues (Goguen 1992, Sauer 1993). Further, some (less able?) analysts assert that if they work through a methodology step by step, working to the letter, rather than the spirit of the methodology, then they will have produced a ‘good’ system. Consequently, one of the prime motivations for the approach developed here was to ensure that the technical and non-technical issues should be explicitly addressed. Such a way of approaching the problem is identified as a systems approach (Ackoff 1974, Checkland 1976, 1981, Churchman 1968, Weinberg 1988). Ackoff (1974), for example, argues that science-based techniques are inadequate to solve the problems associated with large, complex and interacting systems which must be approached holistically using the systems approach. However, he does accept that disciplinary science is useful for independent problems or sub-problems.

In support of this approach, consideration of the problem owner, the problem and its environment, the problem solver, the methodology and the organisational context are recommended when trying to understand ISD (Episkopou and Wood-Harper 1986, Sauer and Lau 1994). The methodology chosen to support a systems solution was SSM (Checkland 1981, Checkland and Scholes 1990). It is recognised that SSM "sees the ISD process as pluralistic, ambiguous and conflict laden" (Lyytinen 1988). Moreover, it was recognised that SSM gave "a better and more complete identification of stakeholders, a deeper insight into the dynamic nature of IS problems and a careful perception of several ... problems which are often ... unnoticed" (Lyytinen 1988). Politically, the Soft Systems Methodology assumes that those involved in the perceived problem situation agree to come to "an accommodation" (Checkland 1998). It also operates as a learning process. The methodology expects no final fixed outcome, although this is possible. For SSM, a particular problem situation (in this case ISD) is examined through an analysis of the perceived corresponding Human Activity System (HAS), which includes both product and process issues.

The organisation and its stakeholders for this ISD need to agree on the quality issues and quality factors that are important. From this, a contingent ISD can be defined. Software Quality Assurance is seen as an element of the monitoring and control processes of this HAS. As well as offering a means to resolve problems, SSM can be seen as a means of describing an appropriate set of activities that should be taking place in ISD i.e. as a metaphor for ISD. Within this framework, a model of software quality assurance (the Contingent Factors Model) based on SSM will be developed, in Section 2.

This Contingent Factors Model and its usefulness have yet to be fully tested. It is argued that any such test will take the form of action research and will be in the form of a SSM intervention. An intervention might well be supported by an instrument or questionnaire to identify basic information concerning that organisation’s approach to ISD. Such an instrument has been piloted (Bennetts et al. 1999).

Finally, in Section 3, this paper concludes that the rich understanding of ISD and its milieu, developed through the use of SSM, should imply greater ability to focus on appropriate organisational goals when using the Contingent Factors Model, together with an encouragement to recognise and address non-technical as well as technical issues.

THE CONTINGENT FACTORS MODEL

The aim is to develop as wide an approach to software quality as possible, through the use of success factors, which is sensitive to the characterisation of software quality attributes used by an organisation and its development personnel.

It was noted that the engineering response to the Software Crisis, while successful in part, had not succeeded to the extent of removing the problem. This was characterised as a failure to fully address human and management factors. These viewpoints can be addressed through Mitroff and Linstone's (1993) representation of business problem solving, which recognises three generic viewpoints in business problems - Technical, Organisational and Personal. This representation can be linked to both SSM and SQA.

The idea of making quality factors (or attributes or characteristics) contingent developed from two main areas. Firstly, each organisation is unique, with its own problems and its own attitudes. For example, Linstone's Multiple Viewpoint approach (Mitroff and Linstone 1993, Linstone, 1985) showed that Organisational and Personal viewpoints were significant in developing a pertinent solution to any business problem. Secondly, it has been shown that even where organisations had identified their view of the nature of software quality, this was not necessarily supported by the developers, who had different goals (Bennetts and Wood-Harper 1996).

The overall model of this paper is represented by the contingent ISD being linked to the GQM model of Basili (1995). This paper has argued that for successful ISD, Technical, Organisational and Personal Viewpoints need to be identified and incorporated into the techniques and processes that take place. The approach taken here is to adapt the general framework of the processes of SSM due to Checkland and Scholes (1990) by recognising that the issue is focused on ISD. Rather than invoking SSM's Logic Stream, where the techniques are attempting to clarify ideas and assumptions, the approach taken is to replace the Logic Stream by a framework from Multiview2. This framework is defined by the dimensions Organisation and Information Technology use and Systemic versus Reductionist issues (as shown in Figure 1). Into this framework we place four generic activities to describe ISD. They are not supposed to be unique or comprehensive but indicative. The activities are defined as: identify the information processing needs of the organisation; develop full or partial information model of the organisation; design the corresponding information system and identify the impact of the information system on the organisation (as shown in Figure 2).

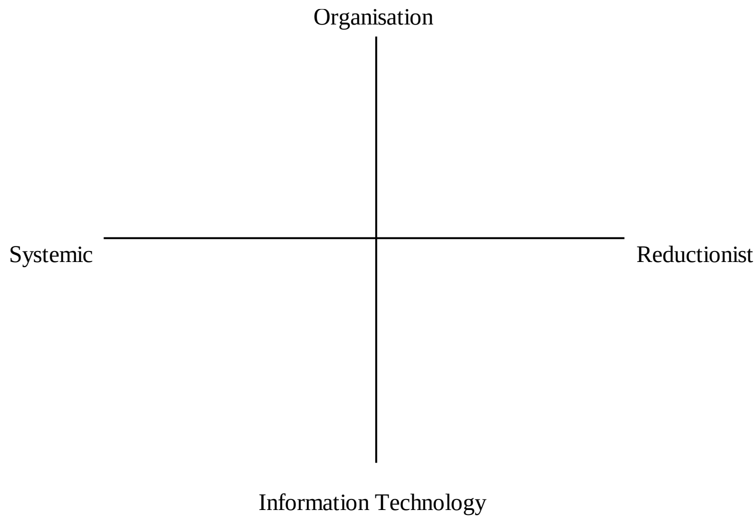

Figure 1: The Multiview2 Framework for ISD (after Bell and Wood-Harper, 1998, Avison, Wood-Harper, Vidgen and Wood, 1996)

In Figure 2, would-be-problem-solvers consider the current situation in the context of historical decisions, information and data. The Logic Stream of this process structure is represented by the framework from Multiview2. The would-be-problem-solvers develop the organisational culture in a way that is culturally feasible. Within the Stream of Cultural Analysis, the generic Analyses One (the analysis of the roles being played by stakeholders in the problem situation), Two (the analysis of the social system) and Three (the political analysis) will have been used to identify the options that were culturally feasible. These decisions will include the business goals and quality factors to be addressed by a specific project in that organisation, at that time. The identified goals and factors now form the interface to the GQM approach, which forms the other fragment of the Contingent Factors Model. The measurements taken during the product development are used to initiate the GQM processes.

The Contingent Factors Model is summarised in Figure 3. As the links between the goals or characteristics and the factors or attributes, for example, are too numerous for them all to be shown. Similarly, no attempt has been made to identify the links between specific factors and sub-factors, or between sub-factors and questions. The Goals, Questions and Metrics of the GQM approach are shown by these three boxes respectively, connected by the many-many ‘crows feet’ symbol. There are only a symbolic five goals, after Miller’s (1956) well known conclusion that most people can do no more than address 7±2 issues at a time, without degradation.

The model of Figure 3 represents the products and processes of one ISD project. In practice, the learning experience achieved through obtaining appropriate metric data, as well as debriefing users and developers, needs to be fed back into the decision making processes, to inform the development of the next project. This represents the standard SSM

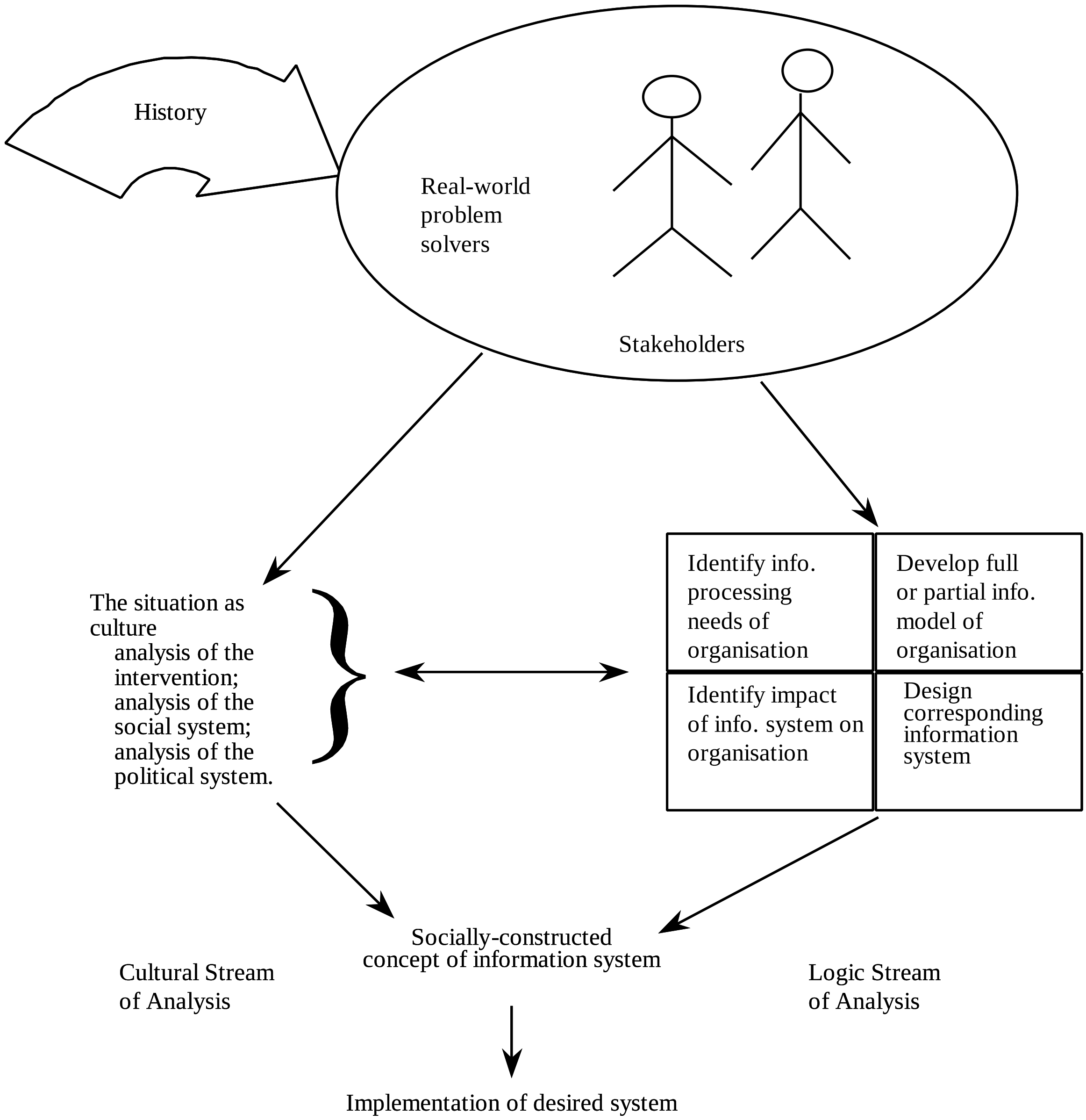

Figure 2: A Framework for the ISD Process

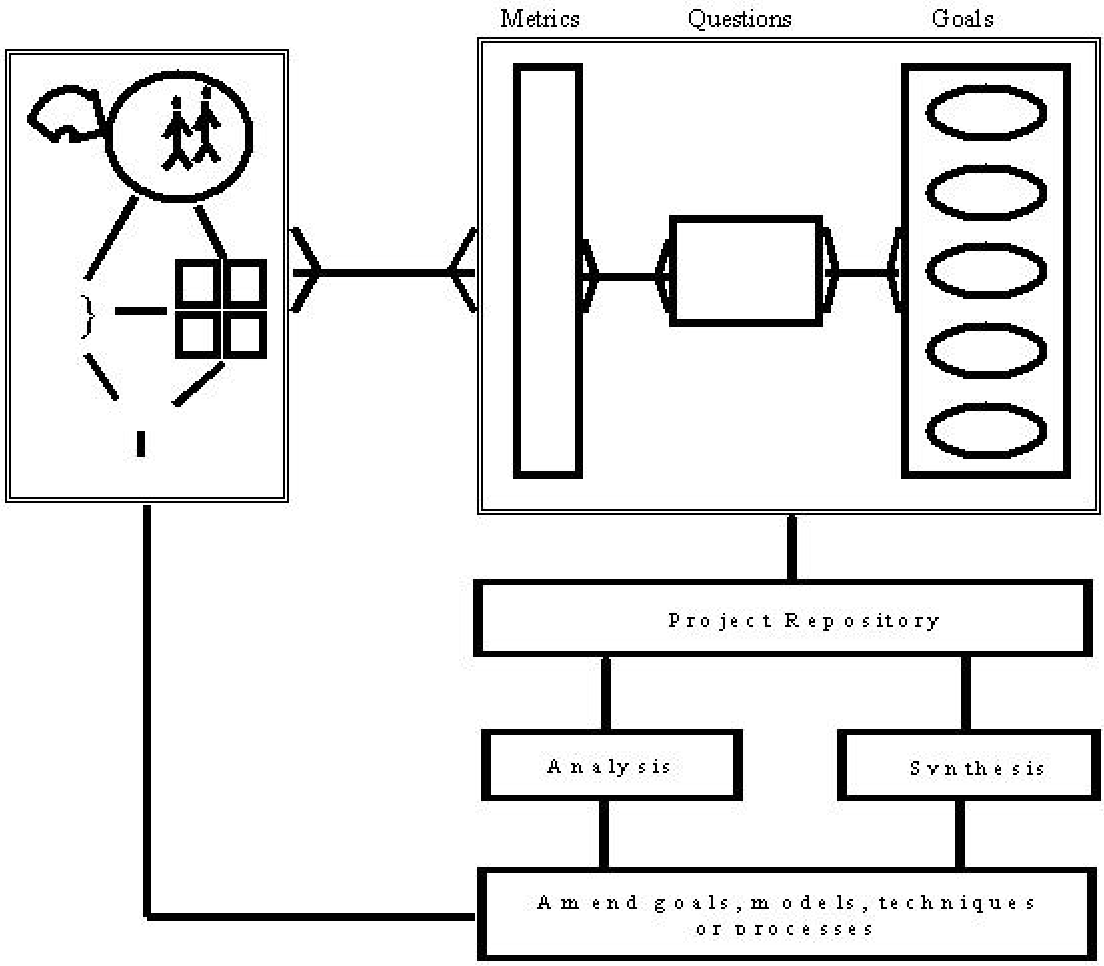

Figure 3: The Contingent Factors Model

process of unlimited iteration. It also, hopefully, represents the sought-after incremental improvement of software quality through SPI. This may be achieved through an improvement process, based on GQM, in which project information is deposited in a repository. The data and information in this repository can be analysed and synthesised to identify potential improvements to perceived goals, models, techniques and processes. Consequently, a feedback loop, through this improvement process, has been included in Figure 3.

It has not been possible to test the Contingent Factors Model yet. Moreover, this model is based on SSM, Multiview2 and GQM, so validation and verification processes should reflect these elements. However, the organisation under investigation may not use any metrics, in which case, little use can be made of GQM. Again, Multiview2 represents a development based on SSM. SSM has been used in well over one hundred interventions (Checkland 1981), with each intervention using a form of action research and, over time, giving rise to changes in SSM. Similarly, Multiview is seen as a form of action research. It is appropriate, therefore, to consider the use of action research to test the model. If Baskerville and Wood-Harper's (1997) characterisation of action research techniques, is used, then it is possible to deduce which technique is most likely to be appropriate. The taxonomy uses four major characteristics - Process Model (Iterative, Reflective, Linear); Structure (Rigorous, Fluid);

Typical Involvement (Collaborative, Facilitative, Expert) and Primary Goals (Organisational Development, System Design, Scientific Knowledge, Training). Hence, use of the Contingent Factors Model in an intervention might be expected to have 'organisational development' as SPI implies change. There is also an argument to suggest that Systems Design should be seen as its primary goal, in that any improvements will be reflected in changes to ISD as a system. If both goals are accepted, the form of action research which addresses both these goals is SSM, only. If only systems design is seen as a goal, then Multiview may be appropriate. If only organisational development is seen as a goal, other characteristics will need to be considered.

CONCLUSION

This paper has introduced the Contingent Factors Model of SPI. This has been developed from a systems theoretic perspective which reflected the concern to explicitly include an examination of the organisational, political and social issues of the moment. The initial stage was to consider SSM, with its Social and Logical Streams of Analysis. The next stage was to focus the Logic Stream on ISD using Multiview2. The Logic Stream is a representation of the process of the technical development of this information, benefiting from the greater understanding of the context as provided by the Social Stream. The achievement or otherwise, of any organisational goals is expected to be measured by appropriate metrics.

The Soft Systems Methodology was specifically designed to assist in the resolution of ill-structured problems. It was also designed as an interpretative approach, allowing insights to be gained about this form of problem situation. The background to this paper clearly indicates that ISD is not well structured. If it was well-structured, it would be very much easier to control. Further, it was shown that politics and human factor issues are critical. Approaches based purely on the science paradigm have difficulty coping with these issues. SSM, however, is formulated specifically to include these issues. Another benefit from the use of SSM is that management is encouraged to recognise the use of SQA, as part of the overall monitoring and control process.

The value of the model is that it encourages a focus (through SSM's Cultural Stream) on issues which might otherwise be largely ignored but which are recognised as causing failure. While not discarding a vision of a future state, as characterised by the identification of goals and the use of metrics to recognise the extent of their achievement, the use of the Contingent Factors Model will offer greater insight into the current state of the ISD environment, giving a clearer idea of what is systemically required and culturally feasible.

REFERENCES

Ackoff, R.L. (1974) *The systems revolution,* Long Range Planning, 7, 2-20.

Ackoff, R.L. and Emery, F.E. (1972) *On Purposeful Systems,* Tavistock Publications, London.

Avison, D.E. et al. (1996) "Multiview2: A Further Exploration into Information Systems Development", in Hackney, R.A. (ed.) *Business Information Systems - Uncertain Futures,* Manchester Metropolitan University, Manchester, 295-303.

Basili, V.R. (1995) "Applying the Goal / Question / Metric Paradigm in the Experience Factory" in Fenton, N., et al. (eds.) *Software Quality Assurance and Measurement A Worldwide Perspective,* International Thomson Computer Press, London, 23-44.

Baskerville, R.L. and Wood-Harper, T. (1997) A Taxonomy of Action Research Methods, Work in progress, University of Salford.

Bell, S. and Wood-Harper, T. (1998) *Rapid Information Systems Development: A non-specialist's guide to analysis and design in an imperfect world (2nd edn.)* McGraw-Hill Book Company, Maidenhead.

Bennetts, P.D.C. et al. (1999) “Identifying Aspects of Software Quality in Organisations - A Pilot Review” in Hawkins, C. et al. (eds.) *Proceedings of the Seventh International Conference on Software Quality Management* British Computer Society, Swindon, 61-72.

Bennetts, P.D.C. and Wood-Harper, T. (1996) “Multiple Viewpoints and Software Quality Goals” in Samson, B. et al. (eds.) Proceedings of 5th Software Quality Conference, University of Abertay Dundee, Dundee, 66-75.

Bennetts, P.D.C. and Wood-Harper, T. (1997) “Soft Systems Methodology: A Metaphor for the Process of Data Analysis” in Stowell, F.A. et al. (eds.) *Systems for Sustainability: People, Organisations and Environments* Plenum Press, New York, 543-548.

Checkland, P.B. (1976) Science and the Systems Paradigm, *International Journal of General Systems*, 8, 127-134.

Checkland, P.B. (1981) *Systems Thinking, Systems Practice*, John Wiley & Sons, Chichester.

Checkland, P.B. (1998) Seminar on SSM, University of Salford, February 5th and 6th.

Checkland, P.B. and Scholes, J. (1990) Soft Systems Methodology in Action, Wiley, Chichester.

Churchman, C.W. (1968) *The Systems Approach* Dell Publishing, New York.

Clegg, C. *et al.* (1996) “The performance of Information Technology and the role of human and organizational factors”, Report to the Economic and Social Research Council, UK, URL http://www.system-concepts.com/stds/clegg.html>.

Dijkstra, E.W. (1976) *A Discipline of Programming* Prentice-Hall, Inc., Englewood Cliffs, NJ.

Episkopou, D.M. and Wood-Harper, T. (1986) Towards a Framework to Choose Appropriate Information System Approaches, *The Computer Journal* 29(3), 222-228.

Goguen, Joseph A. (1993) “Social Issues in Requirements Engineering” in Proceedings of the IEEE International Symposium on Requirements Engineering, 194-195.

Hoare, C.A.R. (1982) *Programming is an Engineering Profession* Technical Monograph PRG-27, Oxford University Computing Laboratory, Programming Research Group.

Linstone, H.A. (1985) Multiple Perspectives: Overcoming the Weaknesses of MS/OR, *Interfaces*, 15(4), 77-85.

Lyytinen, K. (1988) Stakeholders, Information System Failures and Soft Systems Methodology: An Assessment, *Journal of Applied Systems Analysis*, 15, 61-81.

Miller, G.A. (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, *The Psychological Review*, 63(2), 81-97.

Mitroff, I.I. and Linstone, H.A. (1993) *The Unbounded Mind - Breaking the Chain of Traditional Business Thinking*, Oxford University Press, New York.

Naur, P., et al. J.N. (eds.) (1976) *Software Engineering - Concepts and Techniques*, Petrocelli / Charter, New York.

Sauer, C. (1993) *Why Information Systems Fail: A Case Study Approach*, Alfred Waller Limited, Henley-on-Thames.

Sauer, C. and Lau, C. (1994) “The Adoption of Information Systems Methodologies - An Analytical Framework and a Case Study” in Shanks, G. and Arnott, D. (eds.) Proceedings 5th Australasian Conference on Information Systems, 1, 87-97.

Standish Group (1995) The CHAOS Report, URL http://www.standishgroup.com/chaos.html.

Weinberg, G.M. (1988) *Rethinking Systems Analysis and Design*, Dorset House Publishing, New York.

ACKNOWLEDGEMENTS

The authors would like to thank the referees for their constructive comments.

COPYRIGHT