December 2001

Churchman and Stories of Information Systems Development

Peter Bennetts
Cheltenham and Glouchestershire College of Higher Education

Trevor Wood-Harper
University of South Australia/University of Salford

CHURCHMAN AND STORIES OF INFORMATION SYSTEMS DEVELOPMENT

Peter D. C. Bennetts
Gloucestershire Business School
Cheltenham and Gloucestershire College of Higher Education
pbennetts@chelt.ac.uk

Trevor Wood-Harper
School of Accounting and Information Systems
University of South Australia
trevor.wood-harper@unisa.edu.au

Information Systems Institute
University of Salford
a.t.wood-harper@salford.ac.uk

Abstract

Churchman's framework for inquiring systems has been used by authors interested in taking more than just a positivistic approach to information systems development. This framework is used by Churchman to explore the nature of the design process and the philosophical implications arising. Inquiring systems are supported by a set of conditions, which define three generic roles, played in the design and implementation of inquiring systems. These roles are compared with the generic roles of information systems development identified by Hirschheim and Klein. Churchman develops his ideas using the cause-effect model of physical science, which supports the software engineering approach to information systems development. However, this framework does not completely support interpretative approaches to information systems development, such as Checkland's Soft Systems Methodology. The problems found are shown to relate to the measure of performance and how it is implemented. The significance of this analysis comes from the range of environments that information systems development takes place in from safety critical systems to computer-supported co-operative working, not all of which follow the cause-effect paradigm.

Introduction

Software engineers argue strongly that the methods of information systems development (ISD) should reflect the attitudes and approaches of an applied science (Boehm, 1976; Hoare, 1982; Younessi, 1998, for example). In particular, they are characterised as wishing to ignore all issues that do not reflect this overtly rational process (Goguen, 1992). However, Clegg *et al.* (1996) point out that not only do IS developments usually fail, but the reasons for most information systems failures are due to non-technical issues related to the organisation and its culture and politics. This conclusion is supported by the forthright assertion by Lyytinen and Hirschheim (1987) that "IS failure is an endemic part of information systems evaluation." The implication taken here is that this is sufficient evidence to support the use of soft approaches, for example, in order to address this problem.

Hirschheim and Klein (1989) characterise a Universe of Discourse for ISD that covers all known and possible ISD methodologies. Churchman (1971) on the other hand, has developed a characterisation of the design process (which implicitly includes implementation) from the same viewpoint as the computer scientists. Both characterisations involve the use of archetypal role modelling. In order to ascertain the degree to which Churchman's characterisation can be used to support all approaches to ISD, the roles played in different contexts are compared and issues that do not correspond are identified. The expectation is that this exercise will offer insight into the nature of an appropriate framework to support the roles of ISD.

Performing this correspondence will enable us to contribute to the epistemological support of ISD. Churchman (1971) is used by Checkland (1981) to provide the required philosophical support of a Singerian inquiring system for his Soft Systems Methodology. This paper examines the extent to which this is appropriate.

This paper is organised in the following manner. Churchman's (1971) conditions of system design and their supporting philosophical framework are summarised in the next section. In the following section, Hirschheim and Klein (1989) classify all the various forms of ISD, using four idealised scenarios or story types. These two frameworks are both describing ISD through role-play, so the next section compares them. A short summary and conclusion completes the paper.

Churchman's Conditions for a Design System

The relationships are quoted from Churchman (1971) who states them without developing supporting arguments. Some of these conditions are seen by Churchman as necessary conditions, but he does not defend these assumptions. They are all presented here as numbered conditions in Appendix 1.

Churchman (1971) defines his inquiring systems in terms of general systems. These are considered in this paper in the context of one particular class of system - information systems. Further, the "purpose of an inquiring system is to create knowledge which means creating the capability of choosing the right means for one's desired ends". Churchman is interested in "the meaning of a system from the designer's point of view, i.e. in the 'systems approach' to social change" (*ibid*.).

Churchman develops these ideas by referring to an appropriate set of philosophers. The corresponding inquiring system represents "a reconstruction of [that philosopher's] ideas [about the theory of knowledge] in the language of the design of an inquiring system." (*ibid*.). Churchman does not give a rationale for choosing his particular selection of philosophers other than saying that the reasons are partly personal and partly because everything is open to speculation and imagination. His aim is to use those philosophers' ideas to elucidate what is meant by "design" as well as "system".

For Churchman's enquiry into the nature of inquiring systems, there are three generic figures whose existence is necessary (*ibid*.) - the designer, the client, and the decision maker. These figures are archetypes or idealised role models whose actions are described by Churchman (1971) through the conditions given in Appendix 1. Each archetype can be one person or many people or all three can be found in one person (*ibid*.). Further, each archetype is assumed to have a recognised set of interests or values that are to be addressed by any inquiring system. (This will also be true of the archetypes used by Hirschheim and Klein). Conditions 1 to 9, in Appendix 1, are seen by Churchman (1971) to be necessary conditions for "something S [to] be conceived as a system". In this particular case Churchman implies S is an "inquiring system", as the purpose of that text is to develop the concept of inquiring systems to explore the nature of design. Each of the other numbered conditions is extracted from Churchman's (1971) text, where they are conditions about the nature of the role-play in inquiring systems.

One critical issue is that there is a measure of performance that can be used to maximise the value to the client. In the context of ISD, this might be identified by examining the quality of the software produced. Software quality is characterised by ISO9126 as "the totality of features and characteristics of a [software] product, processor or service that bear on its ability to satisfy states or implied needs". This is an all-embracing definition that would allow for both "tangible, and therefore empirically measurable benefits, and intangible, and therefore assessable but not empirically measurable , benefits" (Robson, 1994). Robson goes on to identify various cost-benefit techniques - such as Return on Investment, Discounted Cash Flow, Net Present Value. In particular, she characterises the method of Information Economics, which extends the earlier, traditional techniques to include the assessment of intangible benefits. However, Robson accepts that "scoring each project ... is an inexact science". Consequently, the authors of this paper argue that any measure of performance for ISD will contain subjective elements and hence the maximum value offered to the client by the designer of an IS will depend on who made that measurement. It should be remembered that costing an IS project "has been notoriously bad. An error factor of ten or higher is not uncommon" (Robson, 1994). This is not what the current authors would want from a measure of performance. Indeed, a measure of performance is seen here as something quantitative, being both objective and reliable. Hence, whoever and whenever someone takes a measurement, the answer will still be the same.

When we attempt to match these conditions against the scenarios of ISD, we will assume that all but conditions 8, 14, 15, 16, 18 and 20 can be satisfied, as the remaining conditions have no impact on the role-play. This list of 6 conditions define the role-play between the archetypes and so the role-play within ISD can be used to reflect back on them.

Role Play in Information Systems Development

Within information systems development, Hirschheim and Klein (1989) have identified four distinct paradigms or "stories", each with their own set of actors which are described below, which together cover all known approaches. The four-way framework

follows that of Burrell and Morgan (1979), which was originally set up to describe the 'universe of discourse' for social science or sociology. Hirschheim and Klein (1989) develop their ideas through the concept of four archetypal stories, one for each of Burrell and Morgan's quadrants. However, there is space here to consider only the first two stories. As archetypal stories they are slightly overstated. Each story is identified by a different metaphor for the analyst or systems developer. The first story is developed from a large experiential base, as it is the orthodox or traditional approach to systems development. The second approach is more recent with fewer examples.

In the first story the analyst is seen as a systems expert. Many successful systems have been developed using this approach (Hirschheim and Klein, 1989). There have also been spectacular failures, such as the London Ambulance System. In this approach management are given the role of defining the aims of the system. It is assumed that the specification is as objective as possible. The primary role of the analyst is to be the expert in technology, tools and methods of system design and project management. Politics is seen as irrational. Formality is emphasised, putting less reliance on intuition and judgement. It is assumed that there is one, measurable reality which is the same for everybody. System design is effectively a technical process. These assumptions give rise to some problems for ISD, as is well recognised in the literature (Gibbs, 1994; Scach, 1996; Somerville, 1997) and will not be rehearsed here.

The key actors in this first story are managers (to identify the system objectives); system developers (to construct and implement the system); and users (who operate the system to achieve their work objectives). Information systems are developed to support rational organisational operation and effective and efficient project management. Reality consists of objects, properties and processes that are directly observable in the organisation and the correctness of the system specifications and models can be checked against this. In this story, an organisation's primary goal is to maximise shareholders wealth and management is the only group empowered to decide how this should happen.

Checkland (1981) notices that Churchman's writing is based implicitly on hard systems thinking. Further, Checkland (1981) characterises this approach as goal-oriented (based on the optimisation paradigm). Vickers (1965), argues that this paradigm is "... totally inadequate" to explain what happens in an organisation. However, this "totally inadequate" paradigm represents Hirschheim and Klein's (1989) story type 1 or the traditional approach. A key element of this idealised story is that users tended to be ignored. In practice, the first story tends to favour organisational elites. Hirschheim, Klein and Lyytinen (1995) see this approach as unlikely to persuade those in power to support other interests.

In the second story the key actors are users (those organisational agents who interpret and make sense of their surroundings) and the systems developer (the change agent who helps users make sense of the new system and its environment). In this context, information systems development creates new meaning. Its effectiveness lies in its ability to improve users' understanding of current conventions and meanings. Information systems development proceeds through the application of symbolic interactionism (organisational actors interpret system objectives and specifications and act according to the meaning their interpretation provides for them). The social environment is under continuous evolution - no particular rational explanation clarifies organisational reality. Consequently, the philosophical assumptions are an anti-positivist epistemology (which considers that the search for causal empirical explanations for social phenomena is misguided and should be replaced by sense-making) and a nominalist ontology (which suggests that reality is not a 'given', but is socially constructed and consequently uses social relativism as its paradigm). Thus, in this second story, the business world is explained from the viewpoint of organisational agents who take part in the social process of reality construction.

In Hirschheim and Klein's second story the analyst is seen as a facilitator. This approach has emerged only recently, partly in response to the shortcomings of the first approach, but is in many ways its opposite. It recognises that knowledge about human means and ends is not easily obtained because in reality it is exceedingly complex and elusive. Business does not deal with an objective economic reality, but one that evolves with social laws, conventions, attitudes. No-one has a privileged source of knowledge, each individual sees different aspects. The role of people in shaping reality is not clear. For management, information systems are a part of the continually changing social environment. The distinction between ends and means is fluid and reversible. The role of the system developer is to interact with management to find out what type of system makes sense. There are no objective criteria to distinguish good or bad systems. Any system that has the approval of the stakeholders is legitimate. Systems emerge through social interaction and the continual search for consensus norms and common understandings.

Identifying the Correspondences

The two frameworks which are ostensibly describing the same thing (the various forms of ISD) make different assumptions. It is the interaction of these assumptions which are of interest. To recapitulate: Churchman (1971) recognises three generic figures - the decision maker; the designer and the client. Condition 12 declares that the designer and the client have the same interests.

Further, Condition 24 declares that the interests (or values) of the decision maker are not necessarily the same as those of the other two actors.

Hirschheim and Klein (1989) have identified a different set of generic figures for each story type - managers, system developers and users for story 1; system developers and users for story 2. These actors also have value sets, although their identification is less clear cut than for Churchman. However, as these story types are rooted in the real world, it is possible to make some delineations. Each generic figure was associated with a particular value set.

For story 1, it is usual for the project manager to be in the same department as the system developers. It is therefore assumed that, at a generic level, they have the same viewpoints or the same interests or values. The users' viewpoint is assumed to be different. The essence of story type 2 is co-operation and communication, so that we can assume that the system developer is fully aware of the users' value set and uses it to control the development of the system.

The aim now is to find a plausible set of correspondences between these actors and their various viewpoints. Churchman's propositions are then considered in that context, to see if they are all still true. In order to discern correspondences between these sets of archetypes, we need to assume that if two archetypal roles are combined in one person, both roles have the same value set or set of interests. This might not be entirely accurate in that, in practice, a role player may act differently in the two roles. Conversely, one archetypal role may be played, in practice, by a series of different people. In particular, system development is likely to be carried out by a sequence of experts. As departmental "team players" they are assumed to have the same interests.

There were two reasons for making this assumption. Firstly, the archetypes are "ideal" types and Senge (1990) advocated the ideal situation to be one where everybody was focused on the same goal. Secondly, if we do not make that assumption, then it makes a difference who the expert is. However, the aim of the designer is to measure the client's performance and to maximise the value to the client. Hence, the different experts are expected to arrive at equivalent conclusions, in the sense that the measures of performance are the same.

For story type 1, if we compare the roles played by the archetypes, then we could identify a correspondence where the manager is seen as the decision maker; the analyst is recognised as the designer and the user is seen as the client. This is identified as "Solution A". This might be seen as a natural set of correspondences, but there are some difficulties arising unless we make further assumptions.

Churchman assumes that the decision maker holds one set of values and that the designer and client both have the same set of values, which is distinct from that of the decision maker. However, in Hirschheim and Klein's story type 1, the manager and the systems developer have the same set of values and the users have another. As Mumford (1995) points out - in this traditional (data processing) approach, the values of the information systems developers and managers were to look for efficiency gains by reducing staff numbers and imposing tighter financial and organisational controls. If we take the client to be the staff who will be using the new system, the neglect of the client's attitudes can cause problems when the system is implemented (Mumford, 1995). Indeed, deBrabender and Theirs (1984), quoting studies from 1959 to 1975, suggest that user participation is the main determinant in the successful use of computers. The failure to involve the users represents a failure of Condition 8.

Another way of looking at this form of information systems development is to consider that experience has indicated that the successful designer takes account of office politics (Boland and Day, 1989). Indeed, it is difficult to see how the designer could evaluate the client's value system without considering organisational politics, or taking an interpretative approach which could be outside the assumed philosophical framework. In this environment, the decision maker, as resource provider, has the role of describing what is required of the system. However, in practice, there is certainly no guarantee that the user's values or requirements are going to be considered, merely the decision maker’s.view of what they are, as in Solution B, below.

Consequently, we argue that the client may have a minimal role in the decision making, and hence Condition 14 can only hold if it were to refer to the decision maker's world. In Solution A, the designer's intention is to maximise the value of the system to the decision maker, so Condition 8 will not necessarily hold. As we have shown earlier in this scenario, the user tended to be ignored. Conditions 15 and 16 also need "client" replaced by "decision maker". Most significantly, the designer will not be serving the client's interests and so will not be behaving morally, according to Condition 18. Finally, Condition 20 fails as the designer does not nominate the decision maker. Despite all the above argument, we note that these Conditions could all be satisfied within this Solution A, if the decision maker had the same set of values as the designer and the client; and the users had the same set of values as the manager and analyst. However, if this were true, it would no longer be story type 1 but story type 2.

An alternative solution (Solution B) for the traditional scenario would recognise the systems developer as designer, as before and the decision maker and the client as aspects of the manager. That is, the manager as decision maker decides what the client or

user wants. This would require the role players here to have the same value set. However, and most importantly, the users have been excluded, which potentially causes the problems discussed earlier. Ignoring the users is clearly an unsatisfactory solution, although anecdotal evidence shows it does, sometimes, occur. Such systems which do not have user support have the potential to be undermined by those users (Zuboff, 1988). Further, this is precisely one of the reasons that alternative approaches are suggested by Hirschheim and Klein (1989). On the other hand, this actually fits in with the reality described by story type 1. The users are not made an element of a composite decision maker as their value system is assumed to be different from the manager's, which was precisely why the manager took this initiative. This situation is likely to occur whenever a new IS is to be introduced in order to facilitate downsizing, for example. For this solution, with the decision maker, designer and client not having the same set of values, Conditions 8, 14, 15 and 16 are not satisfied. This is because the decision maker and designer do not necessarily want the same things as the client. Again, Condition 18 will not be satisfied in this situation and so the designer will not be behaving ethically according to Churchman or according to, for example, the BCS code of conduct (Rackley, Betts and Webb, 1996) as the users' rights have not been taken into account. (The BCS code of conduct requires members to "... have due regard to the legitimate rights of third parties".). The designer still has no choice about the decision maker, so Condition 20 still fails.

For story 2, Solution C requires that the designer corresponds to the systems developer and the decision maker and client combine in the form of a consensual user. This situation occurs when the users, as clients, are able to take the initiative and control the project. This will imply that the decision maker and client will have the same value set. Consequently, we have correspondences for story type 2, if assumptions about value sets are correct. If they are, then all of Churchman's Conditions except 1 and 20 are valid. (Businesses are not always teleological and the designer is chosen by the decision maker, not the other way around, as noted earlier.)

Summary and Conclusions

Condition 20 was always in difficulty.

A potential problem recognised was that as human institutions, business organisations are not always rational. To put it another way, organisations do not necessarily follow the cause-effect model of nature. Therefore, in certain situations, Condition 1 might not be true.

A real problem area found in the conditions was the nature of the measure of performance. Attempting to use a measure of performance to define a successful system needs a novel approach to software quality. It should be noted that although the required measure of performance can be conceived, it cannot be identified, at least in this context. Condition 8 implies that constituent metrics will need to be of at least an ordinal scale type in order to be able to order measurements in numeric sequence. Also, the metrics used should be objective and reliable, so that the same value is derived, whoever performs the measurement. However, for the interpretative Solution C, the interest is problem resolution but also, there is an implication that qualitative measures are endemic here and there are legitimate measures which depend on who made the measurement. The implications of Conditions 4 and 5 on the coproduction of a measure of performance, have yet to be identified.

Creating a system which does not further (some) current employees' interests will cause difficulties, unless we argue that the needs of some employees outweighs those of others.

Overall, the two frameworks do appear to be equivalent in ISD, in most situations. However, some problems were identified. In particular, the use of ideal types sometimes meant that some situations appeared artificial. Finally, the analysis has shown that the measure of performance may be qualitative in an interpretative environment and consequently cuts across Condition 17, as the ranking process depends on who does it and who has made the measurements being ranked. To that extent interpretative approaches such as Soft Systems Methodology are not supported.

References

Boehm, B.W. "Software Engineering." *IEEE Transactions on Computers,* December 1976, pp. 1226-1241.

Boland, R.J. and Day, W.F. "The Experience of System Design: A Hermeneutic of Organizational Action". *Scandinavian Journal. of Management.*(5:2), 1989, pp. 87-104.

Burrell, G. and Morgan, G. *Sociological Paradigms and Organisational Analysis.* Heinemann Educational Books Ltd., 1979.

Checkland, P.B. *Systems Thinking, Systems Practice,* Wiley, Chichester, 1981.

Churchman, C.W. *The Design of Inquiring Systems: Basic Concepts of Systems and Organizations.* Basic Books Inc., 1971.

Clegg, C., Axtell, C., Damodaran, L., Farbey, B., Hull, R., Lloyd-Jones, R., Nicholls, J., Sell, R., Tomlinson, C., Ainger, A. and Stewart, T. *The performance of Information Technology and the role of human and organizational factors* Report to the Economic and Social Research Council, UK 1996.

de Brabander B. and Thiers, G "Successful Information System Development in Relation to Situational Factors which Affect Effective Communication between MIS-Users and EDP-Specialists" *Management Science* (30:2), February, pp. 137-155, 1984.

Fenton, N.E. *Software Metrics - A rigorous approach* Chapman & Hall, London, 1991.

Gibbs, W.W. "Software's Chronic Crisis" *Scientific American* September, pp. 72-81, 1994.

Goguen, J.A. "Social Issues in Requirements Engineering", *Proceedings IEEE International Symposium on Requirements Engineering (RE93)* Jan 4-6, 1993, San Diego, CA., pp. 194-195.

Hirschheim, R. and Klein, H.K. "Four Paradigms of Information Systems Development", *Communications of the ACM.* (32:10), 1989, pp. 1199-1216.

Hirschheim, R., Klein, H.K. and Lyytinen, K. *Information Systems Development and Data Modeling - Conceptual and Philosophical Foundations* Cambridge University Press, Cambridge, 1995.

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

Laudon, K.C. and Laudon, J.P. *Management Information Systems – New Approaches to Organizations and Technology 5th edition* Prentice-Hall International, Inc. 1998

Lyytinen, K. and Hirschheim, R. Information systems failures - a survey and classification of the empirical literature *Oxford Surveys in IT*. (4), 1987, pp. 257-309.

Mumford, E. *Effective Systems Design and Requirements Analysis: The ETHICS Approach*; Macmillan Press Ltd., 1995.

Rackley, L., Betts J. and Webb, J. "Conflicts of Loyalty - The Client Versus the User and the Stakeholder", *Values and Social Responsibilities of the Computer Science* (ETHICOMP96), Madrid, 6-8 November 1996, pp. 351-363.

Robson, W. *Strategic Management and Information Systems* Pitman Publishing, London, 1991.

Rosenhead, J. "Introduction: old and new paradigms of analysis", *Rational Analysis for a Problematic World: Problem Structuring Methods for Complexity, Uncertainty & Conflict,* In Rosenhead, J. (ed.). John Wiley & Sons, 1989, pp. 1-20.

Scach, S.R. *Classical and Object-Oriented Software Engineering* (3rd edition) Irwin, Chicago, 1996.

Senge P. *The Fifth Discipline: The Art and Practice of the Learning Organisation*, Doubleday, New York, 1990.

Sommerville, I. *Software Engineering* (3rd edition) Addison-Wesley Publishing Company, Reading, MA., 1989.

Vickers, Sir G. *The Art of Judgement - A Study of Policy Making.* Harper and Row, London, 1965.

Younessi, H. Software process improvement: a multi-dimensional perspective; *Software Process Assessment and Improvement* In Rout, T. (ed.) Computational Mechanics Publications, Ashurst, Southampton, 1998.

Zuboff, S. *In the Age of the Smart Machine - The Future of Work and Power.* Heinemann Professional Publishing, 1988.

Appendix 1

The Conditions for a Design System for Churchman

The necessary conditions:(Churchman, 1971, pg. 43)

1. "S is teleological" (i.e. defined within a cause-effect model of nature).
2. "S has a measure of performance" (a number (*ibid*., pg. 47)).
3. "There exists a client whose interests (values) are served by S in such a manner that the higher the measure of performance, the better the interests are served, and more generally, the client is the standard of performance."
4. "S has teleological components which coproduce the measure of performance of S."
5. "S has an environment - which also coproduces the measure of performance of S."
6. "There exists a decision maker who - via his resources - can produce changes in the measures of performance of S's components and hence changes in the measure of performance of S."
7. "There exists a designer, who conceptualises the nature of S in such a manner that the designer's concepts potentially produce actions in the decision maker, and hence changes in the measures of performance of S's components, and hence changes in the measure of performance of S."
8. "The designer's intention is to change S so as to maximise S's value to the client."
9. "There is a built-in guarantee that the designer's intentions [are realisable (eventually)]."

The designer:

10. The designer has to identify "the client and the decision maker" (*ibid*., pg. 48).
11. "The designer needs to have a theory about his role as well as a theory about the system." The designer must "learn about the system" and understand the influence that he can and should have on the system changes that will be required (*ibid*., pg. 52).

The designer and client:

12. The designer has a value structure identical to that of the client (*ibid*., pg. 47).
13. As an actor in this scenario, the client is "described only in terms of his value structure" (*ibid*., pg. 47).
14. The designer invokes a world in which the client could change whatever was wished "within the bounds of limited resources" (*ibid*., pg. 47).
15. The designer seeks to describe the underlying principles of the client's choices, using a "measure of performance" (*ibid*., pg. 47).
16. "The designer is successful to the extent that he can accurately measure the client's real preferences" (*ibid*., pg. 47).
17. The designer must analyse possible futures by designing and, in principle implementing each of them. The measure of performance is used to "assign numerical values to these possible futures" and hence rank them against each other (*ibid*., pg. 47).
18. "The designer is moral if he serves a client who has a legal or moral right to expect that the system will serve his (the client's) interests and his interests are themselves legal or moral" (*ibid*., pg. 48).
19. If the decision maker's ideas about a system are not seen as "good" by the client, then "the designer's role [is to] try to change the decision maker's value structure" (*ibid*., pg. 48).

The designer and decision maker:

20. The designer is expected to choose the decision maker in a way that will maximise the measure of performance (*ibid*., pg. 52).
21. The designer's ideas about a system are expected to produce changes in the actions of the decision maker and hence changes in the measure of performance (*ibid*., pg. 48).
22. "The decision maker coproduces the future along with the environment, which he does not control" (*ibid*., pg. 48).
23. The environment is defined by what is not changed by the decision maker. Decisions about what will be changed by a decision maker and what will not be changed depends on the decision maker (*ibid*., pg. 52).
24. The decision maker also has a value structure, but it is not necessarily the same as that of the other two actors (*ibid*., pg. 48).