Soft systems methodology

4.1 introduction: Soft systems methodology (SSM) is a systemic approach for tackling real-world problematic situations.[1] Soft Systems Methodology is the result of the continuing action research that Peter Checkland,[2] Brian Wilson,[3] and many others[4] have conducted over 30 years, to provide a framework for users to deal with the kind of messy problem situations that lack a formal problem definition.[5][6]

Soft systems methodology

Soft systems methodology (SSM) is a systemic approach for tackling real-world problematic situations.[1] Soft Systems Methodology is the result of the continuing action research that Peter Checkland,[2] Brian Wilson,[3] and many others[4] have conducted over 30 years, to provide a framework for users to deal with the kind of messy problem situations that lack a formal problem definition.[5][6]

Overview

It is a common misunderstanding that SSM is a methodology for dealing solely with ‘soft problems’ (i.e., problems which involve psychological, social, and cultural elements). SSM does not differentiate between ‘soft’ and ‘hard’ problems, it merely provides a different way of dealing with situations perceived as problematic. The ‘hardness’ or ‘softness’ is not the intrinsic quality of the problem situation to be addressed, it is an aspect of the way those involved address the situation. Each situation perceived as problematic has both ‘hard’ and ‘soft’ elements. The very notion of a problem is contingent on a human being perceiving it as such. e.g. One man's terrorist is another man's freedom fighter.

SSM distinguishes itself from hard systems approaches in the way it deals with the notion of ‘system.’ Common to hard systems approaches is an understanding of systems as ontological entities, i.e., entities existing in the real world. As such, in hard systems approaches when one speaks of a computer system, an information system, a telecommunications system, or a transport system, one refers to these as bounded entities with a physical existence which can be formally described or designed to fulfill a given purpose.

In contrast, SSM treats the notion of system as an epistemological rather than ontological entity, i.e., as a mental construct used for human understanding. If we look for example at a particular organisation as a system, we can describe this organisation as a system to make a profit, or a system to transform raw materials into a commercial product, or a system to provide jobs to the local community, or a system to pollute the environment. Depending on what perspective we take, we will have a very different understanding of this particular organisation.

None of these descriptions is right or wrong, they are merely different ways of understanding what is going on. This requires us to become conscious of our particular perspective and values, and these in turn determine what aspects of the situation we understand as being part of the system of our concern. For instance, if we are trying to understand this organisation as a system to transform raw materials into a commercial product, we are likely to include the providers of raw materials and the customers who buy the end-product in our understanding of this system. However, if we look at the organisation as a system to provide jobs to the local community, we are likely to include different elements such as the local transport infrastructure which allows members of the community to access the organisation. As such, depending on our perspective we draw different boundaries around what we perceive the system to be.

History

SSM was developed from 1966 by a team of academics from the University of Lancaster led by Prof Gwilym Jenkins, and resulted from their attempts to tackle management problem situations using a systems engineering approach. The team found that Systems Engineering, which was a methodology so far only used for dealing with technical problems, proved very difficult to apply in real world management problem situations. This was especially so because the approach assumed the existence of a formal problem definition. However, it was found that such a unitary definition of what constitutes 'the problem' was often missing in organisational problem situations, where different stakeholders often have very divergent views on what constitutes 'the problem'.

SSM has received its fame and recognition through the work of Prof Peter Checkland who joined the team in 1969 appointed as the new Professor of "Commercial" systems and Dr Brian Wilson who had joined in 1966 and ran the action research programme through the University's consulting arm 'ISCOL' from 1970.

SSM lends itself particularly well to dealing with complex situations, where those involved lack a common agreement on what constitutes the problem, and that needs to be addressed. In such situations (e.g. How to improve health service delivery; How to conduct a business in a more sustainable way; How best to deal with youth offenders; or How best to deal with drug abuse), there may exist many different perspectives, values, and beliefs around what aspects of the situation are most important and how to address them. Those various aspects perceived as problematic tend to be highly interrelated; changing one aspect is likely to have knock-on effects on other aspects. It is important therefore to develop a comprehensive understanding of those interrelationships between the various aspects of the problem situation. As a systemic methodology, SSM aims to aid its users in developing an improved understanding through an iterative learning process.

As an offspring of Enid Mumford's 1960's "Participative Approach" (and sometime MSc. External Examiner at Lancaster), stakeholders are likely to reach accommodations – agreements about what changes to the situation the participating parties can live with. The notion of accommodation needs to be distinguished from the concept of consensus. Consensus implies that all the stakeholders fully agree that the proposed changes best serve all of their needs. The concept of accommodation recognises that this is a very rare state of affairs in most real-world situations, and that most of the time individual needs can only be partially met by collective propositions.

The 7-Stage Approach of SSM

The original version of SSM as a seven-stage methodology published in Checkland's "Systems Thinking, Systems Practice"[2] has since been superseded in Checkland's work. However, the seven stage model is still widely used and widely taught because its step-wise nature makes it easily teachable. Most important, the model has a barrier running across it to differentiate stages between the Real World, above the line, and Systems Thinking, below the line; the rigor to the method and a latter day *pons asinorum* for many students,

The seven stages are:

1. Entering the problem situation.
2. Expressing the problem situation.
3. Formulating root definitions of relevant systems.
4. Building Conceptual Models of Human Activity Systems.
5. Comparing the models with the real world.
6. Defining changes that are desirable and feasible.
7. Taking action to improve the real world situation.

The dynamics of the method come from the fact that stages (2) through (4) are always an iterative process. The stake-holders (defined as Client, Actors and Owner) engage in a debate guided by the analyst/facilitator. During this debate various root definitions (succinct statements of appropriate systems) and conceptual models are put forward, modified and developed until a desirable model is achieved by consensus. This model then forms the basis for real world changes.[7]

CATWOE

The Lancaster team proposed several criteria that should be specified to ensure that a given root definition is rigorous and comprehensive. These criteria are summarized in the mnemonic *CATWOE*:[8]

> **Clients** – Who are the beneficiaries or victims of this particular system? (Who would benefit or suffer from its operations?)
> **Actors** – Who are responsible for implementing this system? (Who would carry out the activities which make this system work?)
> **Transformation** – What transformation does this system bring about? (What are the inputs and what transformation do they go through to become the outputs?)
> **Worldview** – What particular worldview justifies the existence of this system? (What point of view makes this system meaningful?)
> **Owner** – Who has the authority to abolish this system or change its measures of performance?
> **Environmental constraints** – Which external constraints does this system take as a given?

This form of analysis clarifies what the user of the methodology is trying to achieve. By explicitly acknowledging these perspectives, the user of the methodology is forced to consider the impact of any proposed changes on the people involved.

Measures of Performance

While specifying the CATWOE attributes, it can be useful to establish criteria by which the system performance can be measured. Three basic criteria useful in every situation are:[9]

- Efficacy (E_1) - indicates, whether the transformation provides the intended outcome

- Efficiency (E_2) - indicates, whether the least possible amount of resources is being used to implement the transformation
- Effectiveness (E_3) - indicates, whether the transformation helps to realize a more long-term goal (i.e. if it fits into a long-term strategy of the system)

Conceptual Models of Human Activity Systems

SSM Conceptual Models of Human Activity Systems (Conceptual Models) are notional, they are not intended to represent what exists but to represent a stakeholder viewpoint.[3][4] This is often misunderstood. Figure 1 is not intended to represent how rice is, in fact, cooked; but how the stakeholders think it is cooked or how they think it should be cooked or how they would like it cooked.

Conceptual Models take the form of bubble diagrams in which descriptions of activities are enclosed in bubbles and the bubbles linked to each other by arrows. The arrows are intended to represent logical dependency. In Figure 1 the activity "wash rice" is said to be logically dependent on the activities "obtain rice" and "obtain water" being performed. This relation of “logical dependency” is transitive, i.e. if *cook rice* is dependent on *wash rice* and *wash rice* is dependent on *obtain rice,* then *cook rice* is dependent on *obtain rice*. This would appear to conform to what is known, in formal logic, as hypothetical syllogism. However, a connection with logic has been challenged and it has been argued that SSM conceptual models are not “logical” in any sense of the word.[10]

In Checkland's work[4][11] Conceptual Models are usually limited to a small number (seven, plus or minus two) of bubbles. Also, in fidelity to Cybernetics, the main activities are always supplemented by bubbles representing monitor and control systems. However, in Wilson's Information Requirements Analysis[3] the Conceptual Models may expand to include hundreds of bubbles, and the monitor and control systems are dropped. While the principal SSM authors show a high degree of similarity in their accounts of the early stages of the method, considerable diversity begins to appear at the Conceptual Model building stage.

Outcomes and applications of SSM

General descriptions of SSM are highly diverse. SSM has been characterized as a learning system, part of a new paradigm for Operational Research and as a front-end for information system design. However, such diversity is to be expected considering that its aim is to address any kind of unstructured "soft" problem in any organizational or social context. SSM functions as a learning system because it facilitates a greater understanding of the problem situation on the part of those concerned. By bringing out the world views (Weltanschauung) of the people involved in the problem situation, SSM can produce various types of result. The problem might simply disappear as the result of a consensus. A fairly unstructured solution might result, such as agreement to adopt a new role for the organization. A third possibility is that the problem becomes structured, in this case a soft problem resolves into an identifiable "hard" problem.[7] SSM has been used extensively in Information Systems Analysis and Design and some information systems textbooks treat SSM purely as a systems analysis and design method (see Curtis[12]).

The results of a survey of SSM in practice were published in 1992.[13] Based on respondents' answers to an open ended question, the following applications were identified:

- **Organizational design**: Restructuring of roles, design of new organizations, creation of new organizational culture.
- **Information systems**: Definition of information needs, creating an IS strategy, knowledge acquisition, evaluation of the impact of computerization.
- **General problem solving**: Understanding complex situations, initial problem clarification.
- **Performance evaluation**: Performance indicators, quality assurance, monitoring an organization.
- **Education**: Defining training needs, course design, causes of truancy, analysis of language teaching.
- **Miscellaneous**: Project management, business strategy, risk management methodology, case for industrial tribunal, personal life decisions.

SSM for Information Systems Analysis and Design

The uses of SSM in information system design are many and varied. Some of the most notable methods are:

Checkland and Holwell

Checkland and Holwell[11] use SSM at the front end of an information systems design project. Their projects have been concerned with the reorganization of an information systems department, the evaluation of information systems and developing information systems strategy.

Their work does not extend into software engineering and is confined to analyzing the scope and facilitating the management of an information systems design project. As such it has been comparatively free from criticism.

Information Requirements Analysis

Information Requirements Analysis[3] (IRA) seeks to identify the information required in a client organization by building conceptual models comprising hundreds of bubbles. These models are used to derive “information categories” and map activity to activity to activity information flows on a matrix known as a “Maltese Cross”. IRA links directly to software design and has application in building transaction processing systems.

Unlike the Checkland and Holwell projects, where the models remain at a conceptual level, IRA seeks to build models for the design of informations systems that can provide information about real world objects and events (such as stock control systems). It has been argued that IRA models do not have the logical power to represent cause and effect and, therefore, an information system built out of them can not represent events in the physical world.[7]

The use of IRA has not, however, been limited to building transaction processing systems. For example, IRA was used in undertaking an audit of an analysis method — Micro-Analysis — for improving effectiveness and efficiency in a particular area of policing known as 'protective services'. The Regional Project Director was tasked with exploring options for collaboration between North Yorkshire Police, South Yorkshire Police, West Yorkshire Police and Humberside Police. SSM was used to develop a reference model relevant to protective services which, together with information categories for each of the SSM activities and conceptual measures of performance, was used to analyse the efficacy of Micro-Analysis by comparing and contrasting information content.[14]

Multiview

Multiview[15] seeks to front end SSM onto established software engineering methods such as SSADM and Information engineering. Multiview builds Conceptual Models and derives Data Flow Diagrams and Entity-relationship Models from them. Multiview links directly to software design and has application in building transaction processing systems.

The Multiview Conceptual Models are not notional and appear to represent things in the physical world. While this obviates some of the theoretical problems found in IRA, it loses some of the advantages of traditional SSM and opens up a set of problems found in other information system design methods[16]

Logico-linguistic Modeling

Logico-linguistic modeling[17] uses logically enhanced Conceptual Models for Knowledge Elicitation and Representation. These models can be expressed in modal predicate logic from which code in the Prolog artificial intelligence language can be derived. Logico-linguistic Modeling has application in Knowledge based system design.

While Logico-linguistic Modeling overcomes the problems in the transition from conceptual model to computer code, it does so at the expense of making the stakeholder constructed models much more complex. It has been argued that the benefits of this complexity is questionable[18] and that this modelling method is much harder to use[19]