Kent Academic Repository

Mingers, John and Rosenhead, Jonathan (2001) *An Overview of Related Methods: VSM, System Dynamics and Decision Analysis.* In: Mingers, John and Rosenhead, Jonathan, eds. Rational Analysis for a Problematic World Revisited. John Wiley & Sons Ltd, Chichester, pp. 267-288. ISBN 978-0-471-49523-9.

Downloaded from
https://kar.kent.ac.uk/3904/ The University of Kent's Academic Repository KAR

The version of record is available from

This document version
UNSPECIFIED

DOI for this version

Licence for this version
UNSPECIFIED

Additional information

Versions of research works

Versions of Record
If this version is the version of record, it is the same as the published version available on the publisher's web site. Cite as the published version.

Author Accepted Manuscripts
If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding. Cite as Surname, Initial. (Year) 'Title of article'. To be published in *Title of Journal* , Volume and issue numbers [peer-reviewed accepted version]. Available at: DOI or URL (Accessed: date).

Enquiries
If you have questions about this document contact ResearchSupport@kent.ac.uk. Please include the URL of the record in KAR. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies).

12 *An Overview of Related Methods: VSM, System Dynamics, and Decision Analysis*

John Mingers and Jonathan Rosenhead

Introduction

Any book is of finite size, and the topics of books are generally selected from an approximately continuous field. Where ever one draws a line round the subject matter, there are subjects on the other side of the exclusion line which could almost as easily have been selected for inclusion. This chapter provides much shorter summaries of some of those methods which are, at the very least, near neighbours of PSMs.

The characteristics of PSMs include – an orientation to group working, a basis in transparent model representation of the problematic situation, and an iterative and interactive mode of working. The extent to which the methods described here (the viable systems model, system dynamics, and participative variants of decision analysis) match these characteristics is a matter for debate – not least between the editors of this volume. However what is clear is (i) that readers who wish to know about PSMs will in many cases also be interested in a further range of methods which bear so strong a resemblance; and (ii) that such knowledge will also be of value since these methods are not infrequently used in combination with PSMs. This topic of the combination of methods will be addressed in the following chapter.

The Viable System Model (VSM)

The viable system model (VSM) is unlike other methods described in this book as it is not in itself a methodology or process for problem structuring or interventions. Rather it is an abstract model or generic blueprint for helping to design the structure of an organization. Its main tenet is that for an organization to be *viable*, that is to be able to survive within a changing environment, it must undertake particular activities and there must be certain relations between them. The model itself is at a very general level and so can be implemented in many different ways. The VSM has been developed after studying how human beings are organized as viable systems, based on the principles of cybernetics

(Beer 1966; Beer 1972; Beer 1979; Espejo and Harnden 1989). It is analogous to an abstract design for a house that specifies what there must be, e.g., cooking area, living area, etc., and specifies important principles of design based, for example, on ergonomics.

Fundamental Principles – Viability, Variety and Cybernetics

Cybernetics is a term coined by Norbert Weiner (1948) to refer to 'the science of communication and control in the animal and the machine'. It is concerned with how complex systems can control and regulate themselves through feedback processes that rely on information and communication. This is central to the question of whether a system – be it organism or organization – can remain viable, i.e., survive, within a particular environment. Viability implies the necessity of a structural connectivity between components that allows it to adapt and become successfully coupled to its environment. This in turn brings in questions of *identity* – what is the 'it' that is surviving? In the organizational context this immediately raises strategic questions such as: What are we? What do we do? What are our boundaries? We should not assume that there are definitive answers to such questions when we build VSM models – rather the modelling process should be seen as establishing some temporarily acceptable conventions that may be useful within organizational conversations.

An organization exists within, and is coupled to, an environment. The organization can be seen as undertaking various activities or operations with respect to the environment – its *primary activities* that produce it, and determine its identity. To survive, however, the organization must be able to regulate these activities and, if necessary, change them. That is, the activities must be *managed*. These are the three essential elements of the VSM – environment, activities, and management, each embedded within the other. The fundamental problem from a cybernetic viewpoint is how to manage *complexity*? Complexity is a tricky concept to define, but it is clearly related to *variety* – the number of states or behaviours that a system can exhibit. There is a fundamental law of cybernetics, formulated by Ashby (1956, p. 207) as the Law of Requisite Variety: 'only variety can destroy variety'. This means that for one system to be able to effectively control or regulate another it must have a similar degree of variety. The problem is clear – the environment will have enormously more variety than the organization, which in turn will have much more variety than the management. The organization can never be aware of, let alone respond to all possible occurrences of requirements of the environment, nor can management ever know every detail of all its employees and activities.

What occurs in practice is that variety is *engineered*, either consciously, or more likely unconsciously. The high variety is necessarily reduced or *attenuated*, while the low variety controller is *amplified*, as shown in Figure 12.1.

Variety attenuation can happen in many ways; perhaps the most common deliberate technique is filtering – only paying attention to totals, averages, yearly figures etc. The greatest *unconscious* attenuator of variety is of course *ignorance*. It is often, also, the most

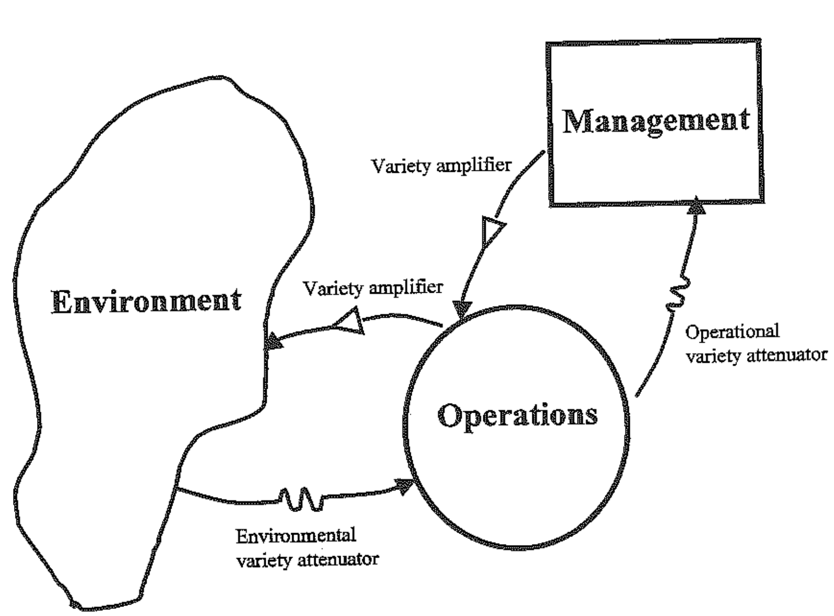


Figure 12.1 Variety attentuation and amplification

lethal. From the other side, an effective organization must amplify its own, and its management's variety – it must generate a richer range of possible actions. In fact, the point is not so much *absolute* variety, but *requisite* variety – there must be a satisfactory balance between the attenuated variety of the environment and the amplified variety of the organization. The system must be designed so as to *absorb* variety with its own. This leads to one of the major premises of VSM – the need for an appropriate balance between central control and peripheral autonomy. Clearly management will find it hard to match the variety of the operational organization, let alone the outside environment, and so it has to allow its operating units autonomy in order to absorb environmental variety. As a final point, computers and information systems are one of the greatest potential tools available to us for both attenuation and amplification yet in many cases they work in precisely the wrong way – presenting us with vast amounts of unnecessary information, and then restricting our range of responses.

The VSM – Systems One to Five

The heart of the VSM is a description of five different functions that need to occur in all viable systems. We have so far distinguished between the primary activities that an organization does, to be what it is, and the management of those activities. These primary activities constitute the System One (*Operations*) of the organization but we need to be very careful in deciding precisely what they are. It is wrong simply to look at a list of Departments or an organization chart – we have to distinguish between the primary activities, *which are viable systems in their own right,* and the secondary activities that support them. A second fundamental premise of VSM is the notion of *recursion.* Viable systems are embedded within viable systems. A university is a viable part of the education system but itself consists of departments that could be viable, and within them courses. The test is, could this activity in principle be taken out of the organization and have its own separate existence? If so, it is a primary activity. If, however, it only exists to support another activity then is in not viable. Thus activities such as accounting, information systems, personnel, and even sales and marketing are generally *not* primary activities since they would have no *raison d'etre* without a product or service.

The concept of recursive or nested viable systems implies that we have to consciously choose the level of our analysis – what Beer calls the *system in focus.* And, at the same time we should be aware of other levels, in particular the levels immediately above and below the system in focus. Figure 12.2 shows how the System One of our system in focus itself consists of several viable systems, each with their own management (the square boxes).

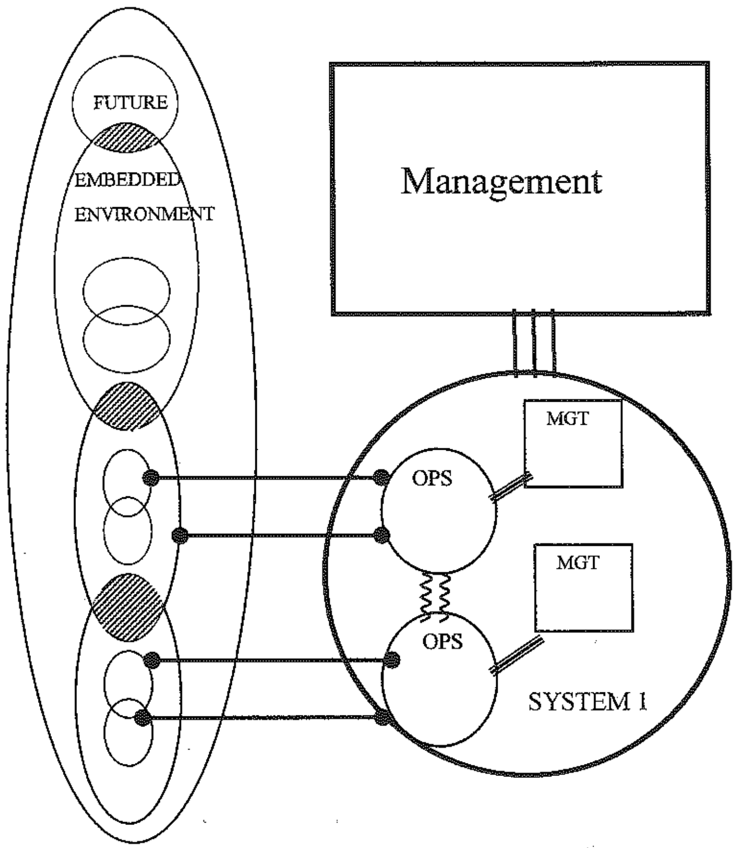


Figure 12.2 System Ones of the System in Focus

Each of these systems will interact with distinct (or perhaps overlapping) parts of the environment and is likely to have a variety of possible interactions between themselves. For instance, the primary activities could be sequential as in a complex production process, or the different stages of an educational system; they could be divided geographically as in different sales or administrative regions of the same organization; they could simply differ in terms of products but with the same population of customers and wholesalers as in a supermarket. In analysing an organization, consider each of its System Ones in terms of all of their interactions with their environments – what are the possible or actual variety attenuators and amplifiers? Do they balance variety effectively?

The different types of structural relations between the System Ones can be recognized within the model but the potential problem that can occur is in co-ordinating or orchestrating their interactions so as to avoid oscillations or clashes. This is the function of System Two – *Co-ordination.* Examples of Systems Twos are: production planning, timetabling and scheduling, project networks, safety codes, and house styles. These are shown in Figure 12.3 as the linked triangles. Such mechanisms do (or should) exist

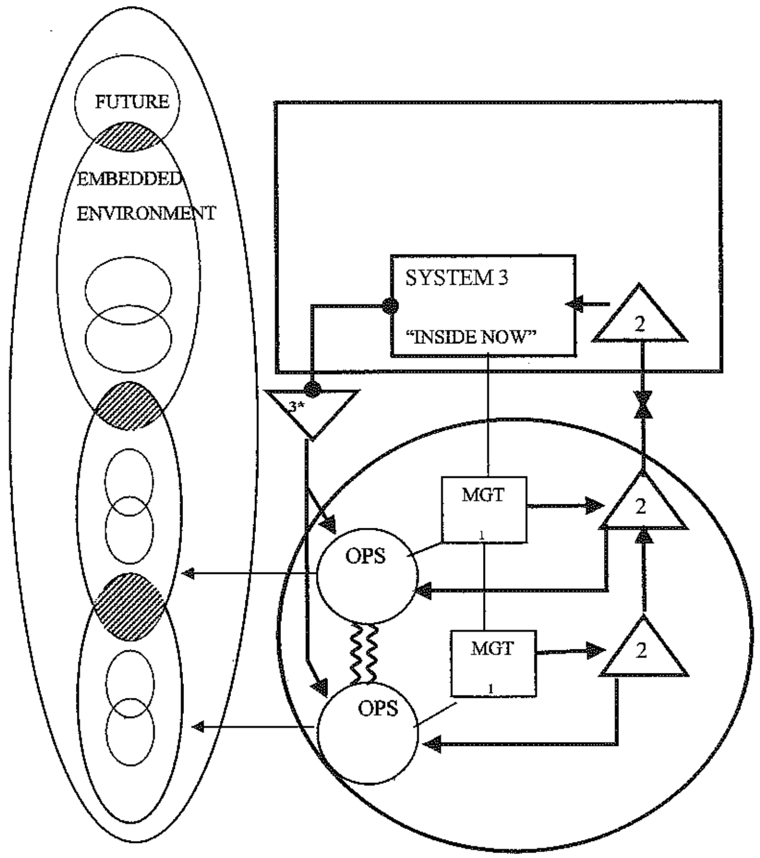


Figure 12.3 System Three – Control

between each System One, as well as at the management level which is able to view the operational system as a whole. The importance of System Two obviously varies with the degree of linkage of the System Ones but it is often under-estimated or even ignored. It should not be confused with management since its purpose is not to control, but to facilitate and smooth.

So far we have not looked into the box labelled 'Management' that deals with the whole of the System One in focus. If we do we find that it is necessary to manage both the internal environment of the organization, the 'inside and now', and the outside environment – especially the future, the 'outside and then'. These two activities are System Three (*Control*) and System Four (*Intelligence*).

System Three is in overall control of all the various System Ones as well as being responsible for the co-ordination function of System Two. Its primary purposes are: to communicate the organizational policy for System One and ensure that it is implemented; allocate resources between the various activities; and monitor actual performance. Of these, the most fundamental is the 'resource bargain' which is at the heart of the balance between control and autonomy. Which activities are to be undertaken and which not? What resources will be made available to support these activities? What are the expectations about performance and how will they be monitored? Once agreements on these matters have been reached, the day-to-day managing of the lower-level activities can be passed down and given their due degree of autonomy. The System Ones will operate within the parameters of the organization – legal, ethical, cultural, environmental – as specified by System Three, and will be accountable to System Three for its results.

These three forms of interaction – the resource bargain, accountability, and corporate identity – will involve considerable amounts of variety attenuation and amplification. This generates a significant potential control problem for System Three. As it stands it has to rely exclusively on information generated by System One to understand what is happening in System One. This information will be of a very attenuated nature and it would precisely go against the whole point of autonomy if System Three were to attempt to regularly scrutinize the day-to-day happenings in System One. The question is then, how can System Three know that it is getting accurate and adequate information from System One? The answer is this it is necessary for System Three to look directly into the operations of System One, but only sporadically, not continually or routinely. This is what is conventionally called *Audit* and within VSM is known as System Three*.

If System Three controls the internal environment, of equal importance is a system to monitor the external environment, especially with regard to the future. It is important to be clear that this function – System Four – is not the same as the interactions with the environment carried out by the various System Ones. These latter interactions will only be partial subsets of the whole environment faced by the system in focus, and importantly they will reflect only the current activities. Many, many organizations have failed because they have not foreseen the changes that make their current operations redundant. There will be, of course, System Fours *within* System Ones at lower levels of

recursion but these will have a more restricted and specific set of concerns. That is not to say that the System Fours at different levels do not communicate with each other and may well thereby learn things of importance for themselves.

System Four (*Intelligence*) is concerned with outside developments, now and in the future, that are relevant to the organization and possible organizational responses to these. This makes it different to the other systems in that it must be aware of, or *have some model of*, the system in focus *as a whole*. This makes it essentially self-referential, for its model must of course include itself. System Four stands at a cross-roads within the organization – it mediates between the outside and the inside, and also communicates important information vertically between Systems Three and One and the policy maker System Five. Its primary function can be seen as one of adaptation – stimulating and bringing about change in response to developments in the environment, as opposed to System Three's function of maintenance and control.

This makes the relationship between Three and Four of primary importance, as shown by the large arrows linking them in Figure 12.4. Too much emphasis on System

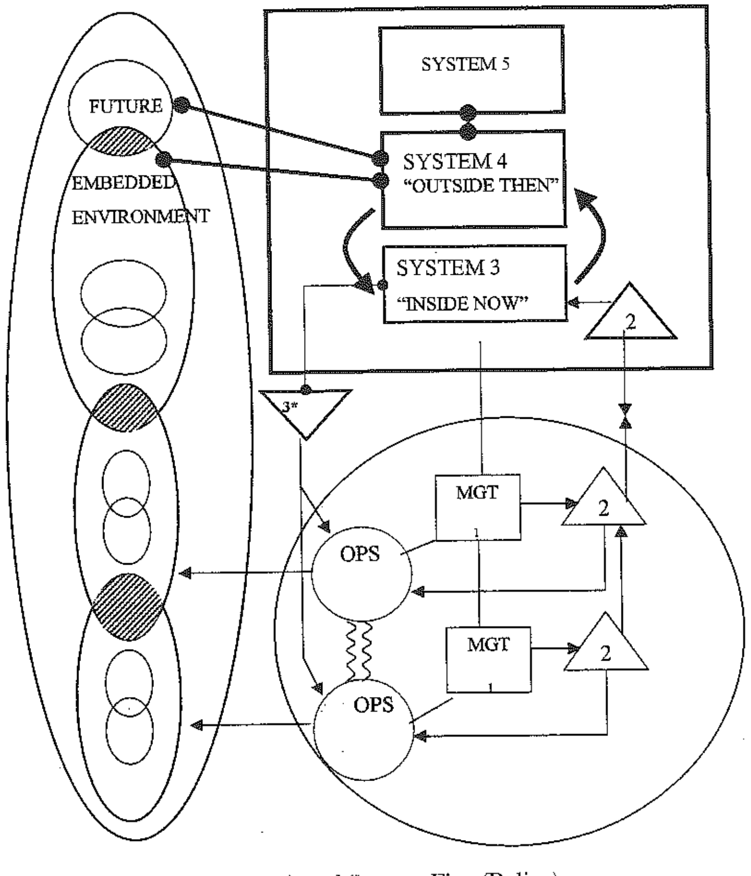


Figure 12.4 System Four (Intelligence) and System Five (Policy)

Four and the future as opposed to day-to-day operations can lead to the collapse of the organization as it is now, while too much concern for internal efficiency can lead to excellent products that have no future.

Finally, we reach the end of the model – the closure of the system – System Five (*Policy* or *Identity*), where the buck stops. System Five, which could typically be the Board (not the chief executive), sets the overall policy and ethos of the organization; it also ensures that the organization has an identity and that this is known and acted upon. This is where we started the model – what are the primary activities that produce the organization and thereby its identity? Beyond that, System Five is necessary to arbitrate in the debates and conversations between Systems Three and Four, and ultimately to determine which of the various futures for the organization will be enacted. It also has a representative function, representing the whole of the system in focus to itself and to outside and wider systems. Finally, it needs to be available to recognize and take action in extreme situations. The organization itself, with its various filters and balances, may well appear to System Five to be functioning unproblematically and so it is necessary for there to be signals, out of the normal channels, that will alert it to the unusual. Beer terms these signals *algedonic*, meaning 'to do with pain and pleasure'. We can see the currently ongoing problems of Marks and Spencer as a failure of System Four to understand developments in the market, and a failure of the algedonic system to alert the board in time.

This brings us to Figure 12.5 which shows the whole of the VSM, including the recursive embedding of the whole model within each System One.

VSM in Practice

The basic model can be used in two ways – for *diagnosis* – by mapping a particular organization on to it to discover weaknesses and problems, and for *design*, in order to construct a more effective structure. It can be used on its own, perhaps within a methodology such as that of Beer (1985) or Espejo, Schumann *et al.* (1996), or is often combined with other approaches such as SSM (see Chapter 13).

All of Beer's books contain many examples and illustrations but the following give case studies of its use by other people:

- Chapters 5 to 11 of *The Viable Systems Model* (Espejo and Harnden 1989) each contains a detailed case study covering, for example, broadcasting, manufacturing companies, and a training network.
- A special issue of *Systems Practice* **3**(3), 1990 is devoted to the VSM.
- The following are illustrations of its use combined with other methods in a multimethodology – Leonard (1997), Ormerod (1998), Gill (1997).

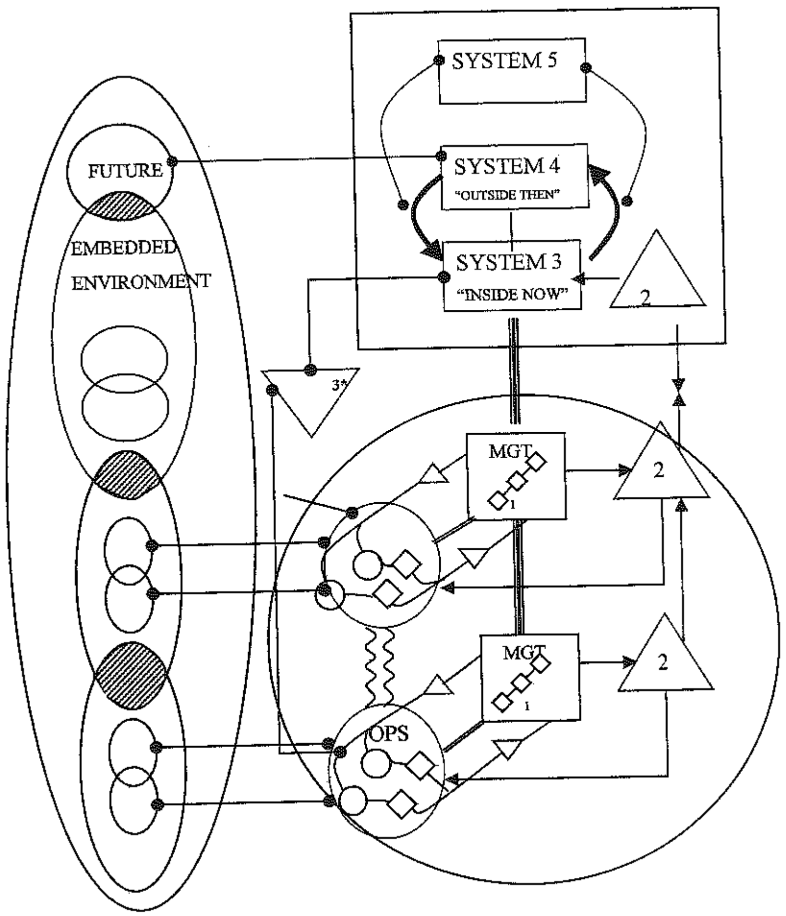


Figure 12.5 The VSM as a whole

When used in diagnosis mode, the following are some of the common problems found within organizations:

- System 1 is not treated as sufficiently autonomous and so cannot deal with its own local variety.
- Systems 2–4 see themselves as autonomous in their own right whereas they are really secondary to system 1.
- The functions of some subsystems are often not performed, especially System 2 – co-ordination, and System 4 – intelligence.
- System 4 is weak because it is seen as a staff (rather than line) function. This may lead System 5 to collapse into System 3 and just undertake control functions.
- System 5 may not be creating a strong enough identity and representing the whole system to its wider systems.

- Information and transmission channels are not appropriate or rapid enough. They may amplify or attenuate variety in the wrong directions – e.g. many information systems.

System Dynamics

The basis of system dynamics (SD) was developed in the late 1950s by Jay Forrester (1961) and was initially called 'industrial dynamics'. It reflected his view that the dynamic behaviour in terms of growth and stability of industrial systems, whether individual organizations, supply chains, or whole industries, resulted from underlying structures of flows, delays, information, and feedback relations. Like the VSM, it drew its inspiration from the pioneering work in cybernetics. Its approach was to develop a mathematical model of the relations between the various components of a system, expressed in difference equations, and then run the model as a form of simulation on a computer. Forrester's work was further developed within wider contexts – e.g., the development of cities (Forrester, 1969) and eventually models of the industrialized world (Forrester, 1973). But it remained somewhat marginal until the 1990s when the advent of good quality graphical software (e.g., *iThink* and *Powersim*), and the popularity of Peter Senge's (1990) work on learning organizations generated a resurgence of interest. There are excellent modern introductions to SD by Sterman (2000) and Vennix (1996).

Fundamental Principles – Counterintuitive Behaviour, Feedback, Dynamic Complexity

One of the most common experiences in trying to manage a situation is that one's actions turn out to make things worse either by generating some form of resistance or adaptation, or by creating a new and often worse problem. Low nicotine cigarettes lead to more being consumed; flood prevention measures like dams often lead to more severe flooding; building new motorways leads to even more congestion and so on. These *counterintuitive* behaviours result from the *systemic* nature of relations – system components are related to each other in multiple, complex ways; cause and effect are not localized but often distant in both space and time; and chains of influence are not linear but circular leading to positive and negative feedback.

In fact, the behaviour of systems is seen as resulting not from the nature of the components themselves but from the relations between components. More precisely, from the interactions between the only two possible types of *feedback* processes, positive (self-reinforcing) and negative (self-correcting, balancing)[1]. Positive feedback occurs when an increase (decrease) in one period leads, through other factors, to a further increase

[1] Note that in SD 'negative feedback' simply means a process where the value of a variable in one period is negatively related to its value in a previous period. In classical cybernetics, negative feedback usually implies 'error-controlled' feedback, that is where the difference between an actual and a desired state is fed back to control a system as in a thermostat

(decrease) in a later period. For instance, an increase in the weapons held by country A leads to an enemy country B increasing its weapons, which pressures A into increasing even more (Figure 12.6a). Conversely, of course, a decrease would tend to lead to further decreases (or at least lower increases). Reinforcing feedback generates a dynamic behaviour of exponential growth or decay. Negative feedback occurs when an increase (decrease) in one period leads to the opposite, a decrease (increase) in later periods. This has stabilizing effects counteracting the initial change. For instance, producing more weapons requires a greater share of a country's wealth, and so will lead to pressure against further increases (Figure 12.6b). Balancing feedback generates a stable dynamic behaviour.

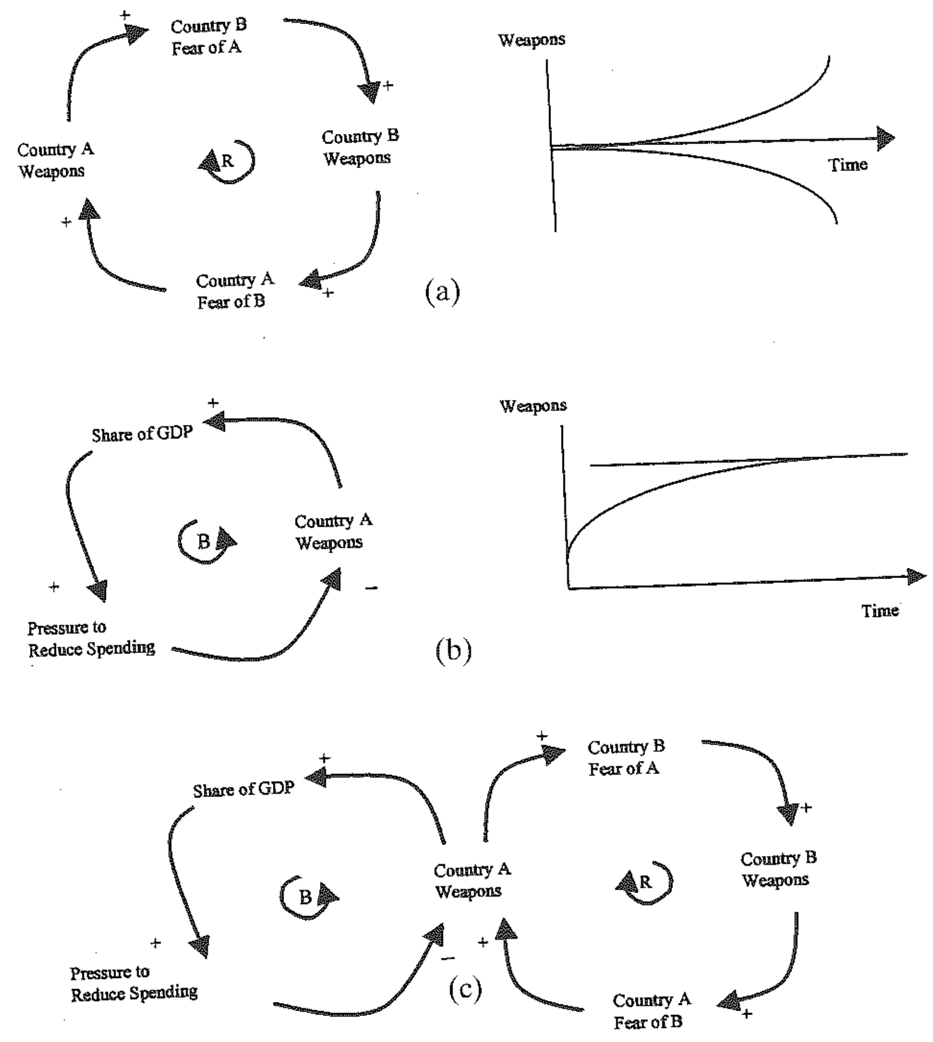


Figure 12.6 Balancing and Reinforcing Feedback

Ultimately, all positive feedback loops must be controlled by negative feedback for there can never be ever-lasting explosive growth, and the *dynamic complexity* of the behaviour of systems is the result of the relative strength of reinforcing and balancing loops over time (Figure 12.6c). Note that there is a contrast between *dynamic* complexity and *detail* complexity; the latter referring to the number of components and relations within a system. Systems with little detail complexity can nevertheless display dynamic complexity, e.g., a magnetic pendulum moving in several magnetic fields, and *vice versa*, e.g., a clockwork watch.

Developing System Dynamic Models

SD models are generally developed for clients within organizations in order to explore and explain some dynamic feature of the situation that is undesirable, or to provide a tool (often called a 'flight simulator' or 'microworld') for training and learning purposes. Table 12.1 shows an outline of the steps in developing an SD model adapted from Sterman (2000), variants can be found in Vennix (1996) and Lane and Oliva (1998). Four points should be made about the process generally:

- It is iterative – models go through continual development, testing, and refinement.
- The process occurs in close cooperation with the client – indeed the process is often one of making explicit the client(s) own *mental model* of the situation. There is, in fact, some debate about what exactly a SD model is modelling. Is it a relatively objective representation of the world, or is it really a model of peoples' beliefs about the world? A range of views are discussed by Lane (1999; 2000).
- Modelling is always embedded within a social and organizational context. There must be continual iteration between the virtual world of the model and learning in the real world.
- In most cases the main point of the whole process is not the construction of an accurate model for *predictive* purposes, for this would be impossible. It is rather the *learning about the situation* generated through the development and use of the model(s).

Problem articulation

This stage is common to all OR interventions and is really what this whole book is about – problem structuring. However, there are aspects particular to developing an SD model. The prime purpose of SD is to be able to explain dynamic behavior in terms of a causal model. This focuses on:

- The time frame – both historical and in the future. It must be long enough to display the behaviour of concern, but not too long so that its detail is lost.
- The boundary to be considered in terms of factors/variables to be included. This

Table 12.1 – System Dynamics Modelling Process. Adapted from (Sterman, 2000).

Problem Articulation	Structuring the problem; determining the main variables, bounding the scope; specifying the time frame; defining the reference mode – 'typical' behaviour
Formulation of Dynamic Hypothesis	Develop maps/causal-loop/influence diagrams of the relations between the factors; identify the main feedback structures; generate hypotheses explaining the behaviour in terms of the feedback processes.
Formulation of Simulation Model	Generate a representation in terms of stocks and flows; estimate all necessary relationships and parameter values; develop a computer model and test for consistency.
Testing and Validation	Comparison with reference mode; robustness under extreme conditions; sensitivity to parameters, initial conditions.
Using the Model – Policy Design and Evaluation	Specify possible scenarios; develop alternative strategies and policies; do what-if analyses; check sensitivity and interaction of policies; use for training.

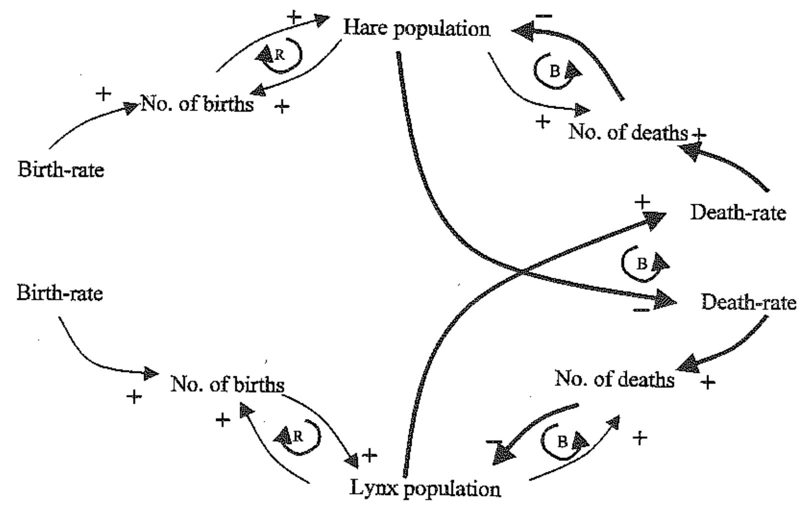
decision is always relative to the particular purpose of the model – what to include and what to exclude is crucial for overall success. There is often a desire for the modeller to create a comprehensive model of everything but this is quite counter-productive – the model should be no more detailed than is necessary for its purpose.
- The 'reference mode' of behaviour. That is the typical behaviour, either unwanted or desired, that the model needs to be able to reproduce.

Formulating a dynamic hypothesis

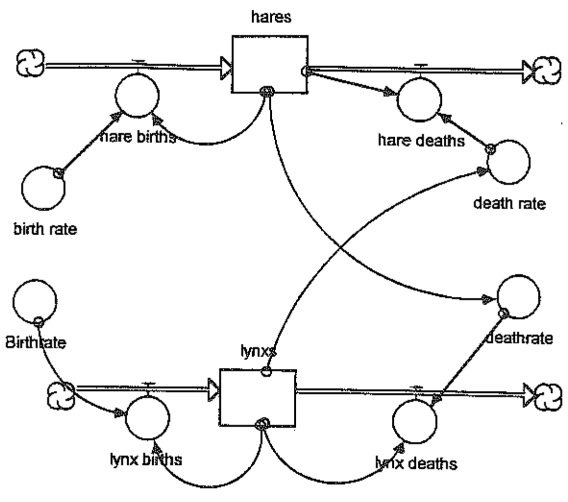
This is in many ways the most important stage of the process as it is where the main causal modelling occurs. The result is a *qualitative* model of the variables and their relationships such that the reference mode dynamic behaviour can be explained *endogenously*, that is purely within the model rather than depending on external factors. The model is, of course, an hypothesis – it is always provisional and subject to development or even abandonment as learning about the situation develops.

There are several tools, especially particular types of diagrams, that are used at this stage. The most common is a causal loop diagram (CLD), also sometimes known as an influence diagram or multiple-cause diagram. Other diagrams, e.g., a model boundary chart or a subsystem diagram are described by Sterman (2000). A CLD consists of factors or variables that are joined by arrows showing the causal links between them (see Figure 12.7a). Each arrow must be labelled with a '+' or '–' to show the direction of causation, that is how the dependent (Y) variable responds to a change in the independent (X). A positive link means that if X increases (decreases) Y will be larger (smaller) than it would otherwise have been. Thus in Figure 12.7a an increase in the number of hare births increases the hare population; an increase in the birth rate increases the number of births. A negative link means that if X increases (decreases) Y will be smaller (larger) than it would have been. Thus an increase in deaths reduces the population.

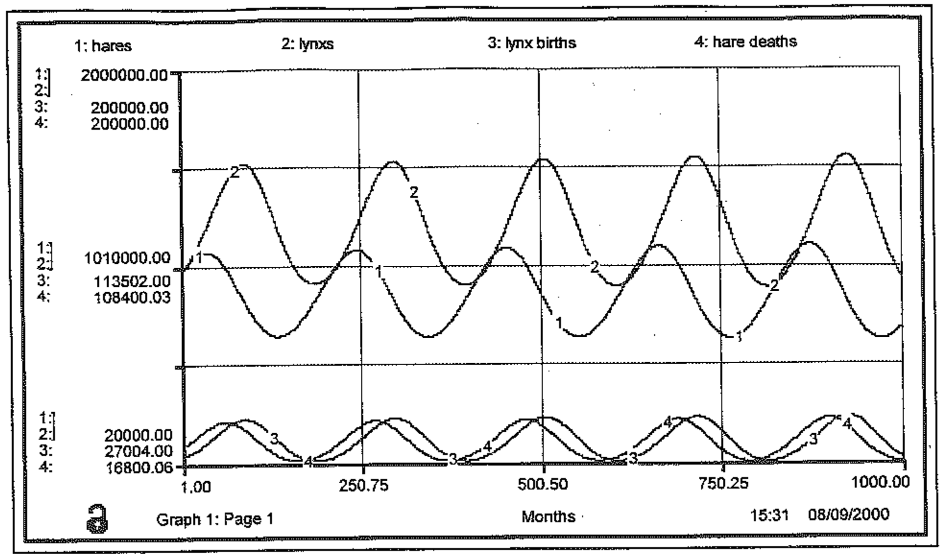
A CLD is intended to identify the causal or feedback loops that involve several variables, as described above. Each loop that is identified should also be labelled as reinforcing (positive) or balancing (negative) – some people use R and B, others + and –. In Figure 12.7a, there is a reinforcing loop between number of births and population size since both causal links are positive. This loop interacts with a balancing one



(a)



(b)



(c)

Figure 12.7 (a) causal loop diagram for competing populations; (b) Stocks and Flows Diagram; (c) typical output from SD software

between population size and number of deaths: as population rises there are more deaths which in turn reduces population. In this classical predator-prey model there are also feedback loops between the two species which need to be traced out. The greater the lynx population the greater will be the hare death rate and this will reduce the hare population. Eventually the reduced source of food for lynxes will lead to more of them dying and start reducing the lynx population which will in turn allow the re-growth of the hare population, thus starting the cycle again. Thus, in this very simplistic scenario the cyclical changes in the size of two competing populations can be generated through the causal structure embodied in the CLD.

The next stage of the process may be to develop the qualitative CLD into a quantitative model that can actually be simulated on a computer. The 'may be' is because many studies do not actually proceed on to a simulation, either because the development of the CLD has generated sufficient learning or because there is insufficient good quality information to produce a useful simulation (Lane, 1994; Wolstenholme, 1999b). Within the context of problem structuring for messy situations the development of an actual computer model is quite rare. One useful result of CLDs is the identification of 'systems archetypes' (Senge, 1990; Lane, 1998). These are patterns of feedback loops and resulting behaviour which are observed very commonly in practice – for example, 'success to the successful' where two activities compete for the same resource, the result being that one gets stronger and stronger at the expense of the other.

Formulation of a simulation model

Clearly we cannot cover in any detail the process of constructing a real SD simulation model – see (Forrester, 1968; Roberts *et al.*, 1984; Coyle, 1996; Vennix, 1996; Sterman, 2000) – but there are two main steps: transforming the CLD into a 'stocks and flows' diagram (SFD), and then estimating all the model relationships and parameters.

Stock and flows, together with information and control, are really the foundations of SD modelling. A stock is the level of a variable at a point in time. This level is changed by inflows and outflows which in turn are controlled by rates of flow or 'valves'. Stocks are not necessarily physical – there can be stocks of memories, beliefs or ideas. They can be thought of as the structure of the system – if a snap-shot is taken at a point in time, what are the important aspects of the system that can be counted or measured? Stocks are shown as rectangles and flows as pipes with valves (Figure 12.7b). The ends of the pipes which are shown as clouds denote the chosen boundaries of the model. The other features are arrows that denote information and control, and auxiliary variables that might be parameters or values calculated from other variables.

Figure 12.7b shows the CLD translated into an SFD; in this case one is very similar to the other. The two stocks are the numbers of hare and lynx, with the inflows and outflows being births and deaths. The causal loops are largely shown by the control arrows. Arrows from the stock of hares go to the flows of births and deaths, while the interactions between the species are shown by the arrows from stock of lynxs to death of hares for example. The diagram itself can easily be created using the icons available in current graphical software but what is important is what is behind the diagram, that is a mathematical formulation of all its relationships. Some will be quite obvious, e.g., $\text{population}_{t+1} = \text{population}_t + \text{births}_t - \text{deaths}_t$. Others will require empirical estimation or expert judgement and may lead to an expansion of the CLD into much greater detail. For example, the lynx/hare death relation might involve specifying the average number of kills per lynx depending on the density of hares, type of terrain, and availability of other types of prey. The software allows relationships to be specified as functional equations or as estimated empirical relationships. Figure 12.7c shows examples of the graphical output that can be obtained from software such as *iThink*.

Testing and validation

In many ways, validating and testing the model is not a separate stage but goes on throughout the process of model building. It can be seen as both a technical process in terms of the internal and statistical reliability of the model, and also as a social process where the prime consideration is the learning and confidence generated for the client (Lane, 1999). We can distinguish between *reliability* and *validity*. Reliability concerns the internal consistency of the model – does it make sense and does it produce consistent results from one run to another? Validity concerns its external relationship – does it reproduce the behaviour that it is supposed to? Some of the main techniques of validation are:

- Replicating the reference mode of behaviour if it was possible to specify one. In practice the model will rarely match the behaviour exactly as it is inevitably a simplification of the world, but it is important to identify and then explain any anomalies. This indeed can be a valuable part of the learning process.
- Testing the model under extreme conditions. It is very easy to develop a model that provides plausible output under typical conditions but which actually has underlying flaws or is inconsistent. One way of discovering this is to test it under conditions that would never actually occur – for instance zero energy input or billions of orders and see whether it does behave as it should.
- Investigating the sensitivity of the model to its parameters, initial conditions, and relationships by varying them in a systematic way. This will show which ones have a particularly significant impact on the results and may therefore be important policy levers.

Using the model

Once the model has been tested and both modeller and client have confidence in its reliability and validity then it can be employed in various ways. One common way is to use it like a flight simulator for training purposes, to put managers in difficult situations and see how they would react and what the consequences would be. It can also be used to explore the consequences of different scenarios for the future. This might just involve minor changes to the parameters or major changes to the whole feedback structure. Equally, the model can be used to test different policies or strategies within the same environmental conditions.

System Dynamics in Practice

There are many examples of CLDs and SD models in the text books already referred to. Also the various software packages that come with a wide selection of ready-built models. Morecroft and Sterman's (1994) edited book contains several case studies as well as theoretical and practical discussion. There is a special issue of the *Journal of the Operational Research Society* (**50**(4), 1999) on SD that also has a variety of case studies as well as more general overviews. The following are recently published case studies: Gonzalez-Busto and Garcia (1999); McCray and Clark (1999); van Ackere and Smith (1999); Wolstenholme (1999a); Bajracharya, Ogunlana, and Bach (2000); Dangerfield and Roberts (2000); and Lane, Monefeldt, and Rosenhead (2000).

With regard to the use of SD within multimethodology, it is probably most often combined with SSM (Cavana *et al.,* 1996; Coyle and Alexander, 1997) and with cognitive mapping, which is closely related to CLD (Eden, 1994; Ackerman, Eden, and Williams, 1997; Bennett *et al.,* 1997).

Decision Conferencing

Decision Conferencing is a workshop-based process which bears, at the very least, a close resemblance to the methods described in the previous ten chapters. Its aims have been described as the achievement of shared understanding, the development of a sense of common purpose, and the generation of commitment to action (Phillips, 1989). It operates in workshop mode employing an independent, impartial facilitator. And it makes use of real-time expert modelling, usually computer-based, to achieve its aims. Most of these are propositions to which proponents of PSMs could happily sign up. (There might be some sensitivity about the word 'expert', with its possible connotation of exclusivity, and about the reliance on computers.) The difference, however, lies in the types of models commonly deployed in Decision Conferences.

Origins of Decision Analysis

Here we need to track back to an earlier period in the history of modelling in support of decision-making. In the 1960's a way was developed of formalizing the choice between decision alternatives when there was uncertainty about future events which affected their consequences (Raiffa and Schlaiffer, 1961; Howard, 1966; Raiffa, 1968). This was most commonly represented as a branching decision tree, and the consequences were taken to be a one-dimensional performance measure (often in practice traded-off and expressed in cash terms). The developments of this approach are known as decision analysis; for full accounts see French (1986; 1989), Watson and Buede (1987), and Goodwin and Wright (1998).

This formulation was extended by Keeney and Raiffa (1976) to allow for there to be multiple criteria, and the approach developing from this innovation became known, not surprisingly, as multi-criteria decision analysis (MCDA) – though multi-attribute utility theory (MAUT) is a more precise name. One of the features of this approach was an interest, not only in the branching tree of decisions, but also in the hierarchy of objectives held by the decision maker. That is, a 'value tree' is constructed, with the most general objectives at the top, each of which is subdivided into components, which may again be subdivided etc. It becomes necessary, therefore, both to weight the components against each other at each level, and to score any alternative action on each of the most basic sub-components of the tree. The formulation may or may not involve establishing the probabilities of outcome, and their incorporation into the calculations. For a full treatment of MCDA, see Belton and Stewart (2001).

The information that is needed for these models can impose considerable demands on the decision-maker, and there are now a wide range of alternative methods which tackle this problem in different ways – see DETR (2000). All of these variants are most often used in the traditional manner, with expert analysts eliciting information, operating on it, and reporting results to clients. The details of their different mathematical formulations are not the issue from the stand-point of this book. What is relevant is the fact that they can in many cases be used in participative mode. This mode is known as 'Decision Conferencing'.

Workshop-based Decision Analysis/MCDA

There is an absence of full-length descriptions of Decision Conferencing – accounts are often a few paragraphs to a few pages either within a description of Decision Analysis/MCDA more generally, or as preface to a case study (Phillips, 1989, 1990; Watson and Buede, 1987). However the general structure of a Decision Conference is quite clear, and in most respects similar to that of a PSM workshop.

The conference itself will last two or exceptionally three days, and be located away from the participants' workplace. Sometimes there may be a sequence of conferences held at intervals. All the key players need to be present, and the conference is run by a team of two or more facilitators. Generally the lead facilitator will meet the chief client in advance to ensure that the issue is appropriate, and to set expectations.

At the conference itself there will be broadly three phases – of formulating the nature of the problem, of model building, and of exploring its implication for decision. Belton and Stewart (2001) call these phases problem structuring, model building, and using the model to inform and challenge thinking.

Belton and Stewart's first phase, problem structuring (sometimes called 'problem framing'), is concerned with the identification of the problem or issue – who are the stakeholders, what are the goals and values, the alternative courses of action and constraints, what are the relevant environment and the prevalent uncertainties? There is an evident convergence here with PSMs, and indeed there is growing experience of the use of PSMs to assist with the framing phase (Belton, Ackerman and Shepherd, 1997; Bana e Costa *et al.*, 2001).

There is no firm line between the second and third phases. There is general agreement that the model is not 'an objectively faithful representation of the problem' (Watson and Buede, 1987), and that its role is not as a finished object to be used to identify optimal solutions. Rather it serves as a vehicle for the ongoing discussion of issues between the parties. Phillips (1990) emphasizes that the model is a rough one constructed rapidly; it is explored in detail only in so far as the differences seem to matter. Modifications are proposed and tried out successively; re-weightings and re-scorings are conducted, which often demonstrate their negligible impact on the model's outputs.

The initial output of a model may not be in agreement with the group's intuitive preferences. In this case both the model and the intuition are challenged, which may surface tacit objectives or generate new options. This process continues until the group accepts the developed model's implications for commitment.

What distinguishes the analytic dimension of different decision conferences from each other is the particular structure of model that is used. What distinguishes decision conferences as a whole from the PSMs described previously is the nature of decision analytic/MCDA models as a category.

As regards differences between models, they may arise from the nature of the problems that they are designed to address, from the process of elicitation and re-processing of information which is adopted, from the software being used, or some combination of these dimensions. Thus, for example, the well-known HIVIEW and EQUITY software packages cater, respectively, for problems of choosing between alternatives, and of allocating resources. Other multi-purpose software explicitly within the MCDA tradition includes V·I·S·A (Belton and Vickers, 1990) and MACBETH (Bana e Costa and Vansnick, 1999).

These factors make for a great deal of variety within the field. However it is what these various formulations within decision analysis/MCDA have *in common* that gives decision conferencing certain aspects that are distinctively different from the PSM workshop process. Evidently the mathematical expression of their problem will not, in general, be comprehensible to the group members. Therefore there is considerable reliance on software that can improve transparency by demonstrating diagrammatically the implications of the formulation, and of any changes to it that are under consideration. The mathematics does its work behind the scenes; group members accept the implications through a combination of the transparency that is achievable, and trust in the computer and in the facilitators.

This feature becomes especially prominent in Belton and Stewart's second, model building, phase. While there are differences in individual style, it is not uncommon for the lead facilitator and the computer and the computer display projected onto a large screen to take centre stage in a way that would be extremely rare in a PSM workshop. Indeed on occasion the technical demands of real-time modelling can be intense. Watson and Buede (1987) suggest that a team as large as four may be needed – one to facilitate, and the others to operate the software, take notes, and assemble summary documents and visual aids. Quite commonly the decision conference takes place in a sophisticated purpose-built facility with room for twelve participants at a circular table, and enclosed by whiteboards, screens with back-projection etc. There is thus a sharp contrast with the ambience (no fixed furniture, low tech, blu-tack) that characterizes the typical PSM workshop for most of the methods described in this volume.

It might appear that Decision Conferencing's reliance on models that quantify objectives and combine them onto a single dimension would place a significant ideological and practical barrier between these two approaches. In practice Decision Conferencing is commonly used in an exploratory and non-optimizing way, so that what

unites Decision Conferencing and PSMs is on balance at least as extensive as what separates them.

References

Ackerman, F., Eden, C., and Williams, T. (1997). 'Modeling for litigation: mixing qualitative and quantitative approaches', *Interfaces*, **27**, 48–65.

Ashby, W.R. (1956). *An Introduction to Cybernetics*, Chapman Hall, London.

Bajracharya, A., Ogunlana, S.O., and Bach, N.L. (2000). 'Effective organizational infrastructure for training activities: a case study of the Nepalese construction sector', *System Dynamics Review*, **16**, 91–112.

Bana e Costa, C.A., da Costa Lobo, M.L., Ramos, I.A., and Vansnick, J.C. (2001). Multicriteria Approach for Strategic Town Planning: the Case of Barcelos, Working Paper 36, Department of Operational Research, London School of Economics, London

Bana e Costa, C.A. and Vansnick, J.C. (1999). 'The Macbeth approach: basic ideas, software, and an application', in *Advances in Decision Analysis* (Eds. N. Meskens and M. Roubens), Kluwer Scientific, Dordrecht.

Beer, S. (1966). *Decision and Control*, John Wiley & Sons, Ltd, London.

Beer, S. (1972). *Brain of the Firm*, John Wiley & Sons, Ltd, Chichester.

Beer, S. (1979). *The Heart of Enterprise*, John Wiley & Sons, Ltd, Chichester.

Beer, S. (1985). *Diagnosing the System for Organizations*, John Wiley & Sons, Ltd, Chichester.

Belton, V., Ackermann, F., and Shepherd, I. (1997). 'Integrated support from problem structuring through to alternative evaluation using COPE and V·I·S·A', *Journal of Multiple Criteria Decision Analysis*, **6**, 115–30.

Belton, V. and Vickers, S.P. (1990). 'Use of a simple multi-attribute value function incorporating visual interactive sensitivity analysis for multiple criteria decision making', in *Readings in Multiple Criteria Decision Aid* (Ed. C.A. Bana e Costa), Springer-Verlag, Berlin.

Belton, V. and Stewart, T.J. (2001). *Multi Criteria Decision Analysis an integrated approach*, Kluwer, Amsterdam.

Bennett, P., Ackerman, F., Eden, C., and Williams, T. (1997). 'Analysing litigation and negotiation: using a combined methodology', in *Multimethodology: The Theory and Practice of Combining Management Science Methodologies* (Eds. J. Mingers and A. Gill), pp. 59-88, John Wiley & Sons, Ltd, Chichester.

Cavana, R., Lee, M., Bennet, J., and Taylor, R. (1996). 'Possum and gorse control on a farm woodlot: a system dynamics approach', *Asia-Pacific J. Operational Research*, **13**, 181–207.

Coyle, R. (1996). *System Dynamics Modelling*, Chapman Hall, London.

Coyle, R. and Alexander, M. (1997). 'Two approaches to qualitative modelling of a nation's drug trade', *System Dynamics Review*, **13**, 205–22.

Dangerfield, B.C. and Roberts, C.A. (2000). 'A strategic evaluation of capacity retirements in the steel industry', *J. Opl Res. Soc.*, **51**, 53–60.

DETR (2000). *Multi-Criteria Analysis: a manual*, Department of Environment, Transport and the Regions, London.

Eden, C. (1994). Cognitive mapping and problem structuring for systems dynamics model building, Dept. of Management Science Working Paper 94/6, Strathclyde University, Glasgow.

Espejo, R., and Harnden, R. (1989). *The Viable Systems Model: Interpretations and Applications of Stafford Beer's VSM*, John Wiley & Sons, Ltd, Chichester.

Forrester, J. (1961). *Industrial Dynamics*, MIT Press, Cambridge, MA.

Forrester, J. (1968). *Principles of Systems*, MIT Press, Cambridge.

Forrester, J. (1969). *Urban Dynamics*, MIT Press, Cambridge, MA.

Forrester, J. (1973). *World Dynamics*, MIT Press, Cambridge, MA.

French, S. (1986). *Decision Theory: An Introduction to the Mathematics of Rationality*, Ellis Horwood, Chichester.

French, S. (1989). *Readings in Decision Analysis*, Chapman and Hall, London.

Gill, A. (1997). 'Managing a Virtual Organization', in *Multimethodology: Theory and Practice of Combining Management Science Methodologies* (Eds. J. Mingers and A. Gill), pp. 153–83, John Wiley & Sons, Ltd, Chichester.

Gonzalez-Busto, B. and Garcia, R. (1999). 'Waiting lists in Spanish public hospitals: A system dynamics approach', *System Dynamics Review*, **15**, 201–24.

Goodwin, P. and Wright, G. (1998). *Decision Analysis for Management Judgement* (2nd ed.), John Wiley & Sons, Ltd, Chichester.

Howard, R. (1966). 'Decision analysis: applied decision theory', in *Proceedings of the Fourth International Conference on Operational Research* (Eds. D.B. Hertz and J. Melese), pp. 59–71, Wiley-Interscience, New York.

Keeney, R.L. and Raiffa, H. (1976). *Decisions with Multiple Objectives*, John Wiley & Sons, Inc, New York.
Lane, D. (1994). 'With a little help from our friends: how system dynamics and "soft OR" can learn from each other', *System Dynamics Review*, **10**, 101–34.
Lane, D.C. (1998). 'Can we have confidence in generic structures?', *J. Opl Res. Soc.*, **49**, 936–47.
Lane, D. (1999). 'Social theory and system dynamics practice', *European Journal of Operational Research*, **113**, 501–27.
Lane, D. (2000). 'Should system dynamics be described as a "hard" or "deterministic" systems approach?', *Systems Research and Behavioural Science*, **17**, 3–22.
Lane, D.C., Monefeldt, C., and Rosenhead, J.V. (2000). 'Looking in the wrong place for healthcare improvements: A system dynamics study of an accident and emergency department', *J. Opl Res. Soc.*, **51**, 518–31.
Lane, D. and Oliver, R. (1998). 'The greater whole: towards a synthesis of system dynamics and soft systems methodology', *European Journal of Operational Research*, **107**, 214–35.
Leonard, A. (1997). 'Using models in sequence: a case study of a post-acquisition intervention', in *Multimethodology: Theory and Practice of Combining Management Science Methodologies* (Eds. J. Mingers and A. Gill), pp. 105–26, John Wiley & Sons, Ltd, Chichester.
McCray, G.E. and Clark, T.D. (1999). 'Using system dynamics to anticipate the organizational impacts of outsourcing', *System Dynamics Review*, **15**, 345–73.
Morecroft, J. and Sterman, J. (eds.) (1994). *Modelling for Learning Organizations*, Productivity Press, Portland.
Ormerod, R. (1998). 'Putting soft OR methods to work: the case of the business improvement project at Powergen', *European Journal of Operational Research*, **118**, 1–29.
Phillips, L. (1989). 'People-centered group decision support', in *Knowledge-based Management Support Systems* (Eds. G. Doukidis, F. Land, and G. Miller), pp. 208–24, Ellis Horwood, Chichester.
Phillips, L.D. (1990). 'Decision analysis for group decision support', in *Tackling Strategic Problems: the role of group decision support* (Eds. C. Eden and J. Radford), pp. 142–50, Sage, London.
Raiffa, H. (1968). *Decision Analysis*, Addison-Wesley, Reading, MA.
Raiffa, H. and Schlaifer, R. (1961). *Applied Statistical Decision Theory*, Harvard Business School, Cambridge, MA.
Roberts, N., Andersen, D., Deal, R., Garet, M., and Shaffer, W. (1984), *Introduction to Computer Simulation*, Productivity Press, Portland.
Vennix, J. (1996). *Group Model Building: Facilitating Team Learning Using System Dynamics*, John Wiley & Sons, Ltd, Chichester.
Senge, P. (1990). *The Fifth Discipline: The Art and Practice of the Learning Organization*, Century Books, London.
Sterman, J. (2000). *Business Dynamics: Systems Thinking and Modelling for a Complex World*, McGraw-Hill, New York.
van Ackere, A. and Smith, P.C. (1999). 'Towards a macro model of National Health Service waiting lists', *System Dynamics Review*, **15**, 225–52. *Systems Dynamics*, John Wiley & Sons, Ltd, London.
Watson, S.R. and Buede, D.M. (1987). *Decision Synthesis: The Principles and Practice of Decision Analysis*, Cambridge University Press, Cambridge.
Weiner, N. (1948). *Cybernetics: or Communication and Control in the Animal and the Machine*, MIT Press, Cambridge, MA.
Wolstenholme, E. (1999a). 'A patient flow perspective of UK Health Services: Exploring the case for new "intermediate care" initiatives', *System Dynamics Review*, **15**, 253–71.
Wolstenholme, E. (1999b). 'Qualitative vs quantitative modelling: the evolving balance', *J. Opl Res. Soc.*, **50**, 422–28.

13 *Multimethodology – Mixing and Matching Methods*

John Mingers

Introduction

Most of this book is concerned with exploring particular, generally 'soft', problem structuring methods. This chapter, however, is not about a single method(ology) but about the possibility of combining together different methods, or parts thereof, within a particular organizational intervention. Different types of methods, such as hard and soft, focus on particular aspects of the very complex world which decision-makers have to deal with. Therefore, employing more than one method in combination will help to address the different levels and dimensions of a problematic situation.

At its simplest, *multimethodology* just means employing more than one method or methodology (I will generally talk of 'methods' but some approaches, e.g., SSM, are referred to as 'methodologies'*) in tackling some real-world problem. For instance, one could be using SSM but feel that some cognitive mapping might be useful in understanding how certain managers are thinking. Or one could use SSM as a whole to gain agreement on desirable changes, and then build a simulation model to help implement them. Or you could do some cognitive mapping and then develop this into a causal-loop diagram and ultimately a system dynamics model. It is often sensible, especially for beginners, to use one main or overall methodology, such as SSM, and then augment it by bringing in techniques from others. Alternatively, one can use several whole methodologies to address different parts of the problem situation. The most ambitious approach is to link together different parts from several methodologies, creating a design specific to the particular situation.

Why Should We Bother?

There are three main arguments in favour of multimethodology. The first is that real-world problem situations are inevitably multidimensional. There will be physical or material aspects, social and political aspects, and personal ones. Different approaches tend to focus attention on different aspects of the situation and so multimethodology is

* Many terms you will come across can be used with different meanings. I have included a glossary at the end to explain how I will be using them in this chapter