The Use of Soft Systems Methodology (SSM) As A Tool For Investigation

Susan Gasson, OR/S Group, *October 1994*
Warwick Business School

1. The Purpose of SSM

Soft Systems Methodology (SSM) was devised by Checkland (1981) and elaborated by Checkland & Scholes (1990) to provide a tool for investigating an unstructured problem situation. SSM is not a system *design* tool, but a tool for system requirements *investigation*. Unlike systems analysis tools, such as entity-relationship modelling or data-flow diagrams, which allow the designer to model how the system should operate, SSM questions what operations the system should perform and, more importantly, why. Thus SSM provides a "soft" investigation (into what the system should do) which can be used to precede the "hard" investigation (into how the system should do it). Figure 1 illustrates the purpose of SSM; it should be noted that SSM does not produce either a set of information system *requirements* or an information system *design*. SSM produces a set of feasible and culturally-acceptable actions which can be taken to improve the problem situation. These actions may be used to produce a set of information systems requirements, but it is more helpful to see them as a set of organisational process improvements, where a process is a set of organisational tasks performed purposefully by a human actor or actors.

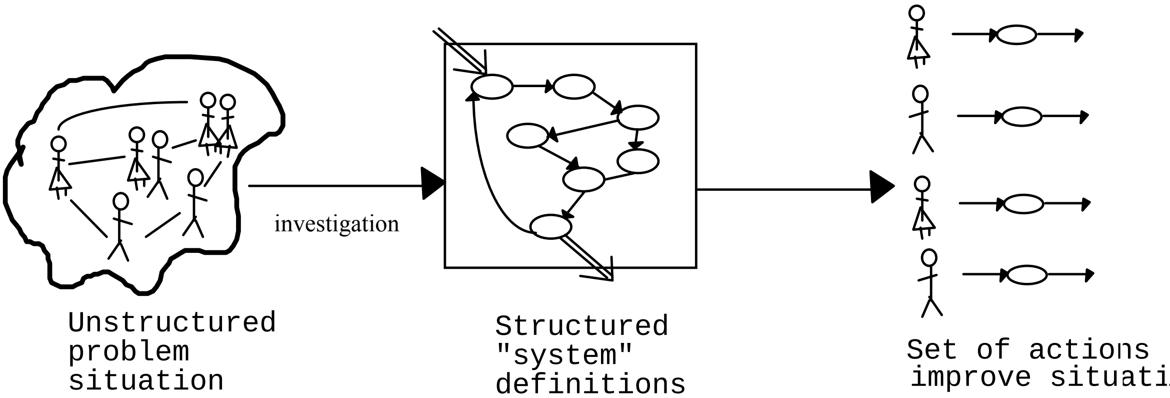


Figure 1: The Purpose of Soft Systems Methodology

Most methods for Information Systems requirements determination place the investigator (the systems analyst) in the role of "expert". In other words, the analyst attempts to understand the organisational situation, determines what computer system functions are required by interviewing potential system users and/or managers, then draws on their expert knowledge of other computer systems to specify a hardware and software "system", based upon their understanding of the organisational situation and tasks. If the analyst's understanding is incorrect or incomplete, this is usually not detected until the system is tested by its organisational users, which may not happen until the system has been installed.

This situation occurs because system users do not normally possess sufficient technical knowledge to understand and question the highly-technical system specification documents which the analyst produces, so any misunderstandings are not revealed until the system is built.

One way of avoiding this problem is to produce a system prototype, where the system designer builds a trial version of the system for the users to try out. Comments and criticisms are fed back to the designer, who changes the system and produces another prototype; this process is repeated until both designer and users are happy with the system's operation. However, prototyping is not appropriate for all development situations (such as very large, complex systems, where each user will only understand a very small part of the system's operations) and also, a great deal of effort in producing and modifying inappropriate prototype systems can be avoided if a more effective investigation is performed into the system requirements in the first place.

The area where SSM is of most use is in information system requirements investigation, where the problem situation is seen as "fuzzy" or ill-defined. In other words, it is not immediately clear what type of system or systems will solve the problems of the organisational work-system.

The advantage of SSM in such cases is that it models explicitly the different value-systems (perspectives) of various people in the organisation, representing different people's differing beliefs about the effectiveness and purpose of the organisation (or their part of the organisation) and their perceptions of current information systems within it. When organisational problems are difficult to define, many problems can be due to work "systems" (for example a department) prioritising one set of organisational objectives at the expense of another set of objectives. For example, a computer-based system for the authorisation of travelling expenses, used by staff in the accounts department may have been designed to reject claims which do not conform to low-cost criteria. Therefore, use of this system will force (or "encourage") staff to use the form of travel which has the lowest cost (e.g. train rather than car) even when the company may lose business by someone not being able to reach a business client before a competitor. The use of SSM for the analysis of requirements for a travelling expenses system would make this conflict explicit, leading to the specification of an alternative system which was able to flexibly approve the expense on the joint criteria of cost and urgency.

2. The Philosophy of SSM

SSM is a *facilitative* method for organisational investigation: the investigator does not model the "system" in isolation, but facilitates (provides support for) organisational actors in their modelling of the organisational "system". The language used by Checkland is aimed at changing the way in which the analyst/investigator views the task of system requirements investigation. In order to understand SSM, any investigator must first understand the reasons for the terms used.

An information system is not seen as a computer system, but as a **human activity system**: the combination of purposeful human activity which may or may not be supported by computer-based technology. The idea of a system is based upon a holistic understanding of human interactions, where the whole is greater than the sum of its parts: by examining the separate functions of a computer-based accounting system, one cannot appreciate the tasks performed by *all* of the people in an accounting department.

The investigator is not engaged in specifying a computer system, or in solving an organisational problem (although the reason for her/his investigation may be in response to a problem perceived by management). Instead, the emphasis is upon the investigation of a **problem situation** - the context of the human activity is as important as the tasks performed.

The people who work in the context of the problem situation are seen as **organisational actors**. This term implies *purposeful* activity on the part of the people engaged in the human activity system - it shifts the responsibility for determining what needs to be done to perform the functions of the system from the analyst to the actors in the human activity system.

The process of SSM is seen as an enquiring process. The investigator is not an expert, dispensing solutions from their vast fount of technical experience, but a **facilitator**, providing support to organisational actors as they themselves define their system's purposes and functions. The role of facilitator necessitates debate between differing perspectives of the system's purposes. This debate is not the same as seeking consensus, which is the implicit aim of "hard" systems analysis. "Consensus" too often means that important objectives of the system are sacrificed to organisational politics, or that the analyst's perspective of the system's purposes is valued above the perspectives of organisational actors because the analyst appears more knowledgeable about technical feasibility. Too often, instead of conflicting system objectives being the subject of an explicit managerial decision by organisational managers, who can make such decisions on the basis of their understanding of organisational objectives, such decisions are made on the basis of an arbitrary decision by the system analyst. SSM therefore, is a means of making such conflicts of interest open to debate and explicit.

Philosophy vs. Application: The Use of SSM In This Course

In a "purist" application of SSM, the input-output transformations and root definitions are derived via mechanisms which involve the stakeholders of the new system, such as a series of workshops, with the analyst as facilitator, where root definitions are hammered out between participants, conflicts of interest exposed, by modelling the activities necessary for these system perspectives and these conflicts resolved by facilitating the stakeholders in mutual agreement of an acceptable set of root definitions. This is the theory. However, this type of process requires an extremely high level of political skill on the part of the analyst/facilitator and also a high level of commitment (and time!) from the stakeholders. **Both of these are unrealistic expectations for your work on this course**, so the use of analyst-derived alternate perspectives is suggested as a way of deriving system definitions. This is described below, in section 3(a).

3. The Process of SSM

The conventional seven-stage model of SSM is shown in figure 2. The seven stages are summarised briefly below.

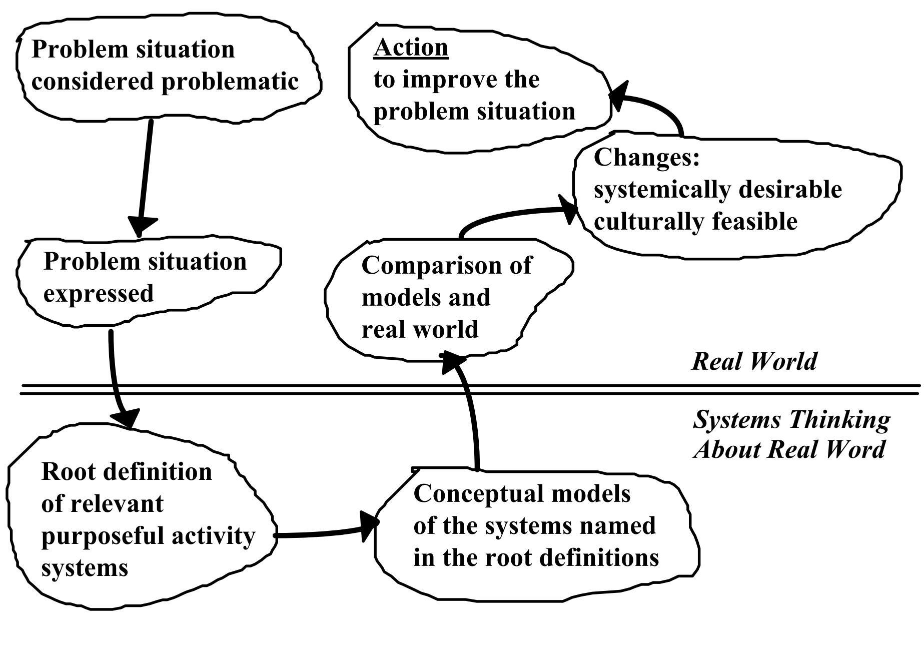
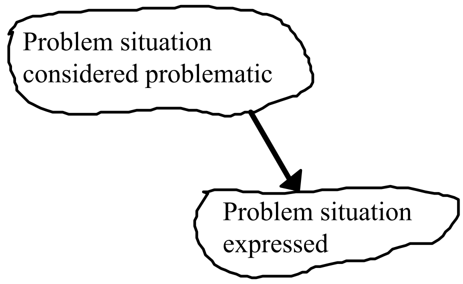


Figure 2: The conventional seven-stage model of SSM

Stage 1: finding out

The investigator(s), referred to by Checkland and Scholes (1990) as "would-be improvers of the problem situation" try to understand, in as wide and holistic a sense as possible, the problem situation context and content. This can be done by the use of interviews, observations and workshops where organisational actors describe their work and the problems which they encounter. It is important to see this stage as a prelude to expressing the problem situation: a means of moving to a state of affairs where the situation is understood reasonably well and is *capable* of being expressed in words and diagrams, thus:



Stage 2: expressing the problem situation

This stage consists of a communication and validation of the investigator's ideas of the problem situation. A variety of tools can be used to achieve this; its purpose is to check that the investigator's ideas about the problem situation are shared by a wide range of organisational actors - some additional tools are described at the end of this paper. The main technique which described by Checkland is the drawing of "rich pictures" - unstructured pictures that communicate everything we can think of about the situation which we are analysing: the main issues, relationships and problems of the organisational context. No drawing skill is required: they normally feature "stick-men" and "stick-women"!

In order to make explicit (visible and open to question) the decision on what to include or exclude, we need to include as much information as possible in order to obtain a "rich" (in the sense of full, complete, wide-ranging) picture of how, and in what environment, our system operates. An example for the traffic warden scheme is attached (figure 3, above) - note that, in order to include all aspects of a situation, you cannot represent everything in great detail, or all of the links between aspects of the situation. Just put in the main links and try to include as many points of view as you can. Then work to a set of slightly more structured diagrams, which you can use to make decisions about system boundaries and system interfaces.

Just drawing these rich pictures makes you (the analyst) think of factors which affect the situation - the more you draw, the more you are stimulated to new ideas by what you have drawn. If you are performing a SSM analysis with a client, ask them to draw pictures. Don't feel that you have to limit the number of pictures - draw as many as you want, to get a full picture of the situation. Peter Checkland suggests that you draw three different pictures, showing different aspects of the situation:

- the *intervention* (analysis process) and your role in it: why are you performing the analysis? for whom? with whom? what does your client want to achieve from the analysis (this is *very* important: what are the goals of your system?) - this information informs your decisions about what to leave in and what to exclude from your system
- the *social context*: who are the people involved in the situation being analysed and how do they relate to each other?
- the *political context*: who holds power? over whom? how is power exerted? how is it resisted? is the person "in charge" the person who makes things happen?

These three diagrams are useful if you are analysing a structured organisational situation. However, don't draw them too early in the process; try to let your rich pictures be as unstructured as possible, to start with, so that you can examine **what is not there** at present, as well as what *is* there. Remember that SSM is a facilitative method: try to persuade organisational actors to draw pictures of their part of the organisation (a good way to get them started is to draw part of a picture yourself, then ask them to fill in the gaps).

Stage 3: Deriving Root Definitions Of Relevant Systems

The purpose of this stage, according to Checkland (1981) is to **name** the system. This is seen as important because, by naming a thing, you define *exactly* what you mean by your understanding of it. This understanding is defined with sufficient precision to enable other people - the client of your analysis, or people who will be affected by the changes which you are proposing - to understand how you are defining their system of work and to contribute to, or perhaps challenge, your definition of it.

The root definitions of your system are derived in two stages: by deriving input-output diagrams, which reflect different perspectives of the same system, then by using these as the basis for a precise "Root" definition of the system as seen from each perspective.

(a) Input-Output Diagrams

This stage imposes some structure onto the analysis, by producing a set of transformations which achieve the purposes of your proposed "system". While wanting to make the system as inclusive as possible, it is important to limit the number of transformations which you will analyse (or you will never complete the analysis!). A useful set of transformations is between 5 to 7, so at this point you make explicit (i.e. discussed with your client or decided by your client) decisions as to what you include and what you exclude from your system.

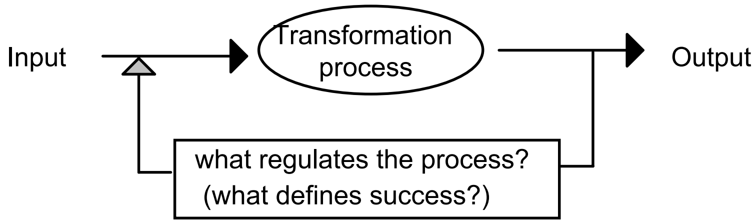
The easiest (and most effective!) way to derive input-output transformations is to interview a representative sample of people from your organisation. This could include:

- the most senior manager who will speak to you
- two junior managers from different functional departments (normally the one who invited you to work with him/her and another one from a totally different department)
- four potential I.T. system users: preferably from different departments, definitely performing different jobs;

this gives you a total of seven perspectives. Each person will have a raison d’être within the organisation - a purpose which they think it is more important to achieve than any other. These purposes are what you are attempting to model in the input-output transformations. Try showing your transformations to the people from whom you derived them and ask if they would define things differently.

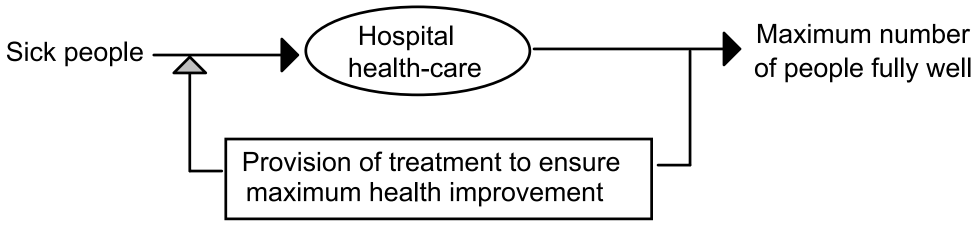
When selecting input-output transformations for your new system, try to think holistically. You are trying to define a set of transformations which will achieve the aims of all the people to whom you have spoken. This is not easy, but try not to junk good ideas (just because a single person gave you two main transformations does not mean that you have to discard one of them) and try not to let the total number of transformations get out of hand! Remember that *any* analyst, however objective, colours their findings by their own understanding of the organisational situation and their own biases about what is important, so try to state your understandings of the situation (try a problem diagram, described below) and try to state your assumptions.

To derive an input-output transformation, you need to define the following variables:

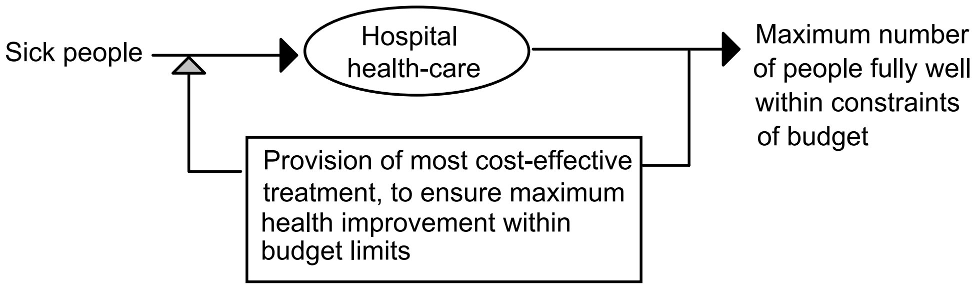


(a) input to a work-process
(b) output from that work-process
(c) the transformation: the work-process which gets you from the input to the output
(d) how you measure success, in achieving the transformation.

For example, the "system" which defines hospital health care could be defined from the view of the nurses, in maximising patient health-improvement. The transformation which they might define would be:



However, if you spoke to the administrators, who have to answer to the local health authority, their transformation might be:



This example illustrates why we use the input-output transformations to derive alternative system definitions. Both sets of people would define their "system" as one to provide hospital health-care. If you were an analyst, you might not speak to the nurses, who have a different definition of success (and a different output from the transformation!) to the administrators. Or you might have internalised (from media interviews with nurses and your own feelings about the NHS) the first perspective and might miss what the administrators were saying to you about the importance of budget constraints. In either case, your system would only contain a subset of the functions necessary to support <u>both</u> sets of people in their work. There are, of course, conflicts in the two definitions of the system. However, rather than pretending these do not exist, your role as analyst is to make them *explicit*, analysing the trade-offs (i.e. incompatible activities, when you model your activity diagrams later on) then have your client decide what their priorities are, rather than you assuming the role of God and deciding their priorities for them!

(b) Root Definitions

The Root Definition consists of **naming** the system which supports each transformation. Checkland's argument is that, until you have put a name to something, you cannot possibly understand its function or purpose. The Root Definition "names" the system in a structured way, which makes sure that you, as the analyst, understand what the system is going to do and how it is going to do it! A common problem with this stage is that, if you do not try to be precise in naming the system, you become confused as to who is performing the activities of the system and what those activities should be. The important thing is to examine each transformation from **one** perspective (this means having *one and only one* set of actors!): you can happily repeat the exercise using another perspective, to see if you can integrate the two (sometimes this may not be possible, for example, how do you integrate the needs of drivers in transformation (1), in the worked example below, with the needs of pedestrians?). If you cannot integrate competing perspectives, you must take a decision about *whose perspective will have priority* - this is where your **client's** objectives come into play. In every change, there are winners and losers; your job is to make it explicit (and therefore your client's decision, not yours) who loses and who wins!

When deriving a Root Definition, the CATWOE model is used, to ask the following questions:

Customer:	who is the system operated for?
Actors:	which *single* group of people will perform the activities involved in the transformation process?
Transformation:	what *single* process will convert the input into the output? Remember that the input and output must be those at the <u>system boundary</u>, *which should be the same for all transformations defined*.
Weltanschhaung: (this means world-view, in German)	what is the view which makes the transformation worthwhile? (this has a lot to do with how you define success, but also states why the transformation process is being performed at all)
Owner:	who has the power to say whether the system will be implemented or not?
Environment:	what are the constraints (restrictions) which may prevent the system from operating?

When deriving Root Definitions, you often need to cycle around, redefining inputs and outputs, then trying to derive a definition from the input-output which you have defined. A common problem is that of being too sloppy in defining terms. For example defining a transformation which represents the UK Traffic Warden scheme in terms of "parking system" is meaningless. It tells us nothing about what the process is which gets us from cars parked in inappropriate places to cars parked in appropriate places. Similarly, defining the input-output as "illegally-parked cars" and "legally-parked cars" tells us nothing about how illegal and legal will be interpreted by different people involved in the system - you need to either define illegal in your Weltanschhaung (e.g. W: "It is desirable that parking in congested streets is made illegal") or define the input and output more tightly (i.e. Input = cars parked in congested streets; Output = no cars parked in congested streets). By being **precise** about terms, *everyone involved in the system understands exactly what is involved* - what activities need to take place and how the system's *success* can be measured.

If you examine the two root definitions derived in the worked example, given at the end of this document, you will see that *these describe completely different systems*. These are definitions which involve a completely different set of goals and activities. **This is *why* we use SSM to analyse a situation**: in a conventional systems analysis, it would be very easy to skimp on the analysis by using input-output diagram (1) to define our system. All the other transformations may well be *intended* by the Government, but if they are not *stated explicitly*, they may well be left out of the eventual system because no-one remembers to include them!

Some transformations may be <u>mutually exclusive</u>, so in some cases an explicit decision must be made by the client of your analysis as to which transformation is going to be "included" in the system analysis at the expense of which other transformation (remember, you do not make these sort of decisions; if your client is to buy into your solution, they have to decide such issues). Other transformations may be <u>complementary</u> to each other, by using this technique, you ensure that you do not miss essential system functions which are not needed by the "main" perspective on the system.

Stage 4: Deriving Conceptual Models

Deriving a conceptual model is a method of analysing the activities which need to take place in order to clearly define what the actors need to do in order to achieve the transformation. Do *not* include activities performed by **anyone** other than the one group of actors whom you have named in the root definition.

List the activities needed to achieve the *objectives* of the system and number them, in the order that they are performed. Remember to include activities which <u>monitor</u> the system and feed-back results. (If you do not monitor whether the system is operating successfully, how do you know whether the system is working properly?). In your input-output transformations, you stated what defines success for this system. Now, it is important to ask “what do my actors need to do, to measure that success?”, in order to derive your monitor and feed-back activities.

Deriving a conceptual model involves two steps:

(a) listing all activities required to achieve the root definition of the system
(each activity should begin with a <u>verb</u>)

(b) graphically relating the activities together, with monitor and feedback activities.

Stage 5: Comparison of Conceptual Models and Real World

The purpose of all this activity is not to just draw pretty pictures, but to provide a solid set of prioritised recommendations for what changes need to be made to existing activity systems (which is a prerequisite for defining what information systems are needed to support those changed systems!). The conceptual models derived can be compared with the real world in a numbers of ways:

- the activities can be considered individually, with each activity compared to real life for its effectiveness and its links to other activities
- activity diagrams (like conceptual models, but for real-world activities) can be drawn and compared to the conceptual models.

Whichever method is used, the intention is to derive a list of process changes: changes to work processes and activities which are necessary in order to move towards the system modelled in the conceptual model.

Stage 6: Analysing Feasible and Desirable Change

An overview of the recommended changes can be gained by the use of activity-linkage diagrams. These diagrams (described below, section 5: Additional Tools) are used to determine how the “top-level” systems link together; they link the transformation-processes defined for each system’s Root-Definition into an integrated “organisational system”. Information-flows between systems and system-dependencies can be observed here; this process gives the “big picture”.

The purpose of this stage is to gain some input from the organisational stakeholders: managers, shareholders, customers of the organisation, those people who will be affected by changes to the existing system and those people who will be involved in implementing changes. Of course, it is not usually feasible to interview a representative sample of all of these people; but the minimum which should be done is to speak to managers and those affected by proposed changes, to elicit their opinion on what their priorities are and what they consider feasible or infeasible and why. Remember that they have more expertise about what makes their organisation work than you have!

Stage 7: Taking Action

This stage involves the "politics" of the intervention. The actions predicated by the previous stages of analysis must be implemented in a way which avoids upsetting too many people, while still achieving the objectives which you set yourself, for the change.

Remember that no change can take place without the commitment of a powerful project "sponsor" - someone who is relatively senior within the organisation and someone who is committed enough to "push through" changes, when organisational politics would otherwise prevent them. Additionally, change is always easier if it is "owned" by the people whom it affects - they must buy into it (in the same way that the stereotypical 1960s wife is shown in films convincing her husband that to buy a new washing-machine was his idea, not hers!).

4. Summary: The Purpose of SSM Analysis

The purpose of performing a Soft Systems Methodology is to define what a system **is** and what it **does**, before trying to design that system. Soft Systems Methodology (SSM) provides a set of tools to define a **human activity system**, *not* an information system. By using SSM, we can include multiple perspectives on what the system is and does, in order to make our decision as to those which we want to include and those which we want to exclude. It is important to remember that SSM analysis is looking for "ideal" systems - a set of activities which represent the *most effective* ways of achieving a set of goals, not the *existing* ways of achieving them. **The purpose of SSM analysis is to suggest ways in which existing systems of activity may be changed, not to represent those existing systems on paper!**

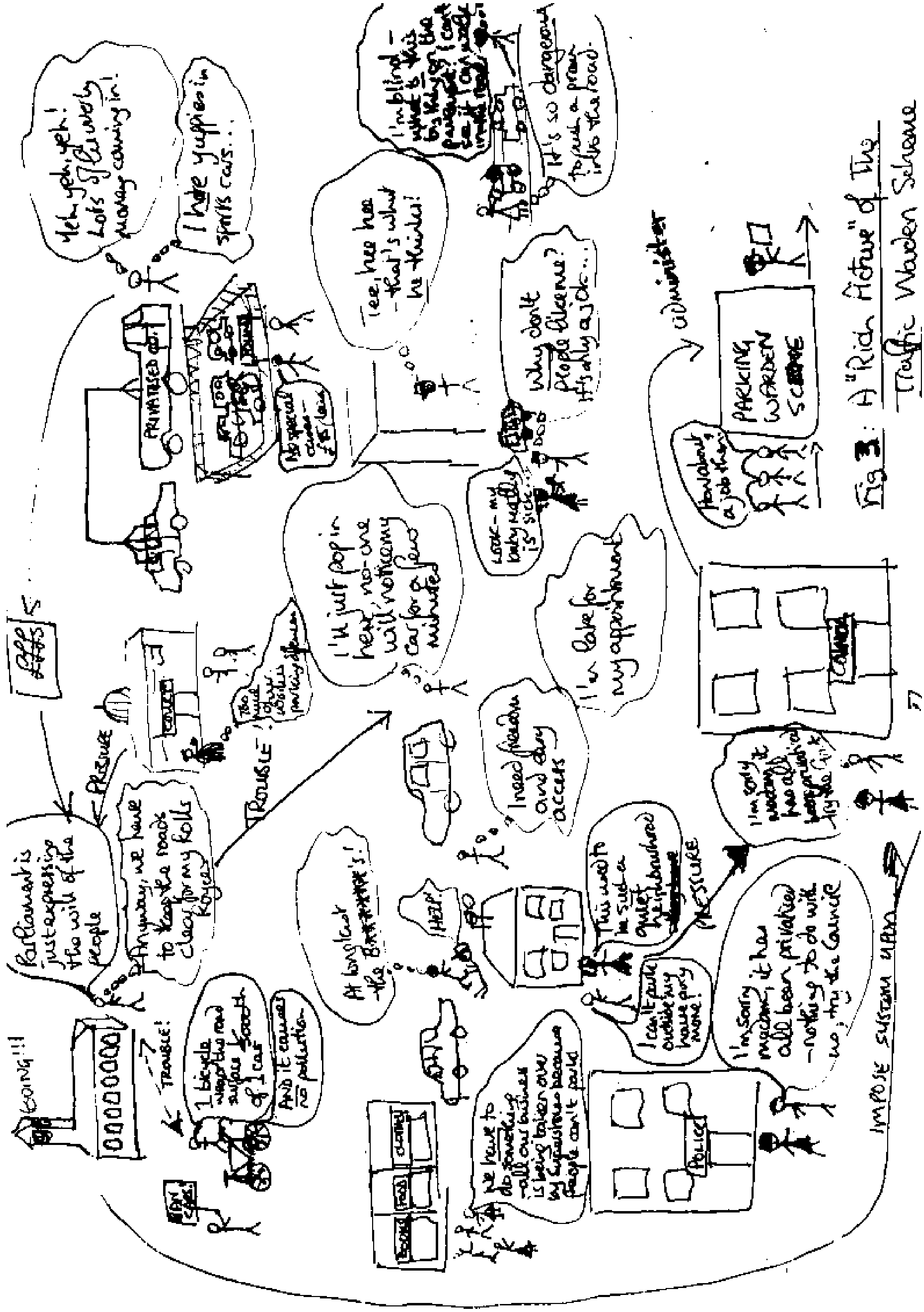


Fig 3: A "Rich Picture" of The Traffic Warden Scheme

5. Worked Example: The Traffic Warden Scheme

Stage 1: finding out

Ideas are exchanged about how the traffic warden scheme works in the UK. Some Traffic Wardens were asked what they thought of the scheme, as well as a sample of drivers, cyclists and pedestrians (i.e. users of the "system" and people affected by it).

Stage 2: expressing the problem situation

"Rich pictures" were drawn of the scheme: a sample picture is given in figure 3. Note that the point of these pictures is not to represent a "system" but to represent everything that the analyst knows about the system, *in as unstructured a way as possible* (so that the existing system structure does not influence the form of the system which you derive), then to progressively structure the picture, to reflect the problem context as it is. The various representations are then compared and used to stimulate lateral thinking about possible forms of a solution, or used to inspire questions to be asked of "system" participants about the purpose of their "system" of activities, or used to analyse the effect of **alternative system boundaries**.

Stage 3: Deriving Root Definitions Of Relevant Systems

(3a) Input-Output Diagrams

A set of input-output diagrams is given in figure 4. Some of these transformations may seem contentious, but try thinking about them from the perspective of the "owner" of the complete national traffic warden scheme: The National Government. I have tried to imagine what their objectives would be and therefore to devise a set of transformations which National Government officials would choose to leave in the "system". These diagrams are most useful if the input relates to the output by a simple transformation - if you cannot put a simple name to the "process" by which the input is transformed to the output, then try another input or output, or try splitting the input-output diagram into two stages. I also find it helpful to scribble the Weltanschhaung under each input-output diagram: the purpose of each transformation (i.e. the view which makes the transformation worthwhile).

Figure 4: Some Possible Input-Output Diagrams:

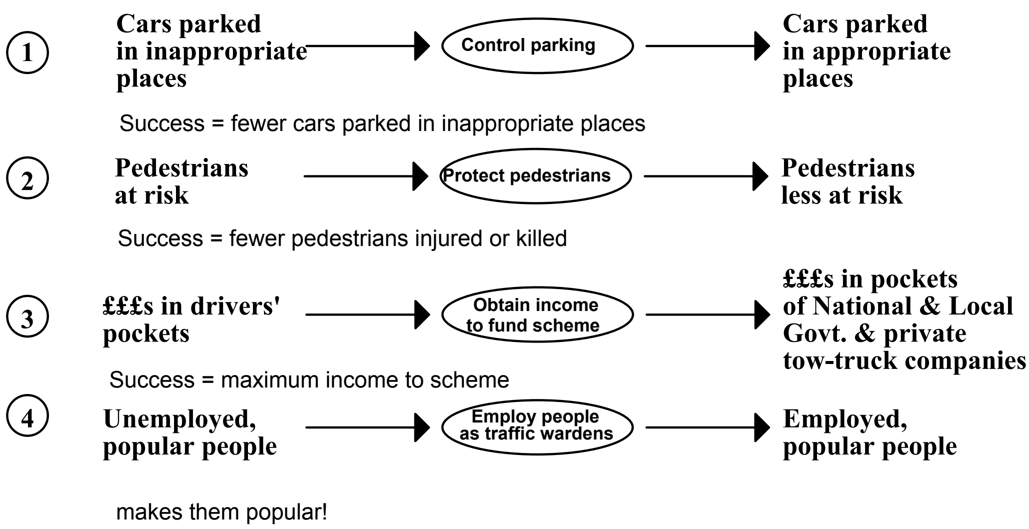
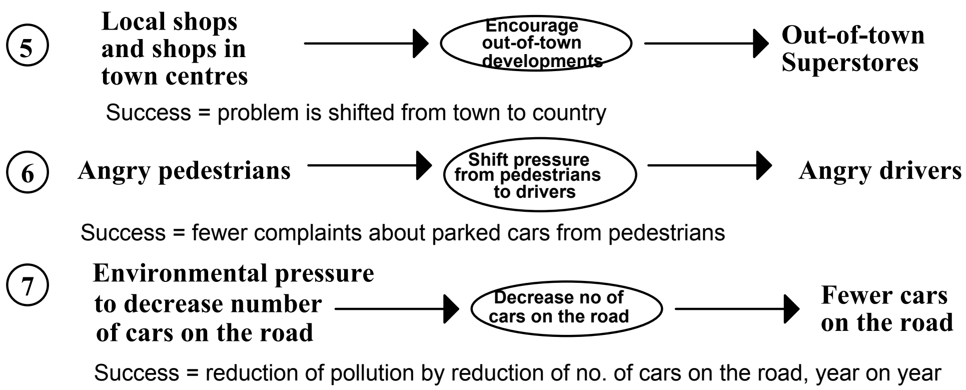


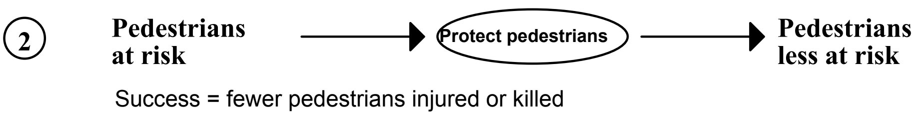
Figure 4 (continued): Some Possible Input-Output Diagrams:



(b) Root Definitions

Two Root Definitions for the Traffic Warden scheme have been derived below; the transformations have been chosen as an illustration because they *describe completely different systems*. These are definitions which involve a completely different set of goals and activities. **This is *why* we use SSM to analyse a situation**: in a conventional systems analysis, it would be very easy to analyse the scheme by just using input-output diagram (1) as the basis for our system and ignoring all the other purposes of the scheme.

Figure 5a: Root Definition for Input-Output Diagram (2):



C: Pedestrians

A: Traffic wardens

T: Punish parking which puts pedestrians at risk

W: Pedestrian safety should have priority over driver convenie

O: National and Local Government officials

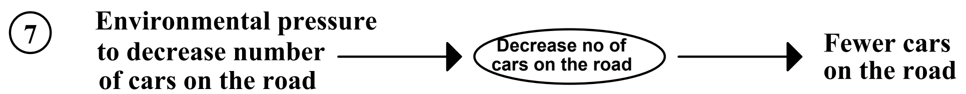
E: Power of drivers' political lobby-groups and available resources for parking regulation

Root Definition:

A system owned by both National and Local Government officials, where Traffic Wardens punish parking which puts pedestrians at risk, on behalf of pedestrians, because pedestrian safety should have priority over driver convenience, as far as the power of drivers' political lobby groups and the availability of resources for parking regulation will allow.

Monitor: number of pedestrians killed or injured.

Figure 5b: Root Definition for Input-Output Diagram (7):



Success = reduction of pollution by reduction of no. of cars on the road, year on year

C: **Environmental lobbyists (the "public"?)**

A: **Local Government Officials**

T: **Make driving less attractive than public transport**

W: **The number of cars on the road is directly related to environmental degradatic**

O: **National Government Officials**

E: **Available public funding and the power of the Drivers' lobby groups**

Root Definition:

A system owned by National Government Officials, where Local Government Officials make driving less attractive than public transport on behalf of Environmental Lobbyists among the public because the number of cars on the road is directly related to environmental degradation, but limited by available public funding and the power of drivers' lobby groups.

Monitor: reduction in road pollution, year on year.

Stage 4: Deriving Conceptual Models

I have chosen to provide a conceptual model for system (7), in figure 4. The Root Definition is given in the preceding paragraph.

The list of activities which are perceived as necessary are:
1. Determine what factors make driving more attractive than public transport
2. Assess what action can be taken to affect those factors by Local Government Officials
3. Take those actions
4. Measure the number of people who transfer from cars to public transport
5. Measure the impact upon the environment of that transfer
6. Report to the public on the results of 4 & 5

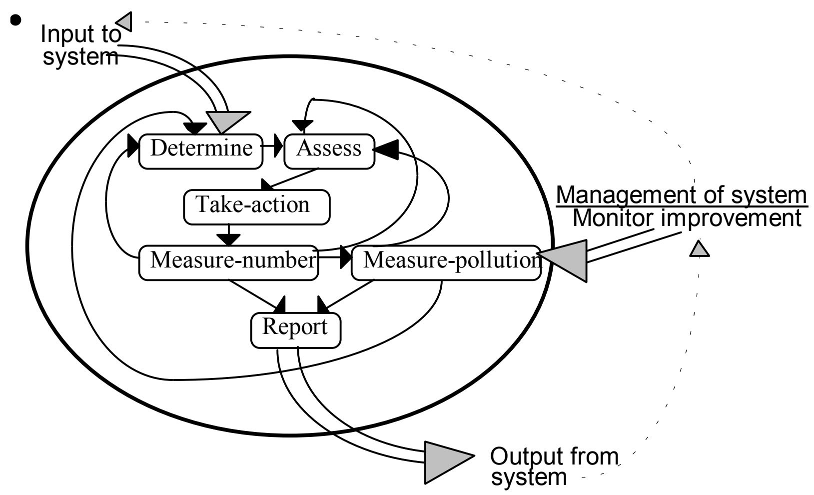


Figure 6: Conceptual Model for System of Input-Output Diagram (7)

Monitoring System Performance Via The Conceptual Model

Note that this system embodies **two** monitoring mechanisms: one of these is internal to the system and is achieved by the feedback of information between activities in the system, which allows the system of activities to regulate itself. The other is achieved via monitoring the definition of success for the system as a whole and is an external, management mechanism, which may be used to monitor the success of the system by people outside of the system. Both of these are equally important and both are vital to the success and continued existence of the system.

Stage 5: Comparison of Conceptual Models and Real World

The method used to illustrate how this stage might be performed is to use an activity grid for the detailed analysis of each activity (see Additional Tools section, below).

1.	Determine what factors make driving more attractive than public transport	Lobby groups (e.g. AA, RAC) ensure activity performed effectively at present **Link missing from activities 4 & 5**
2.	Assess what action can be taken to affect those factors by Local Government Officials	Performed effectively
3.	Take those actions	**May be inconsistent** - need consistent guidelines nationally
4.	Measure the effect of those actions (e.g. the number of people who transfer from cars to public transport)	Measurements attempted but data only used locally. No attempt to perform this activity at a national level.
5.	Measure the impact upon the environment of that transfer	No attempt to perform this activity (politically unacceptable to Govt.)
6.	Report to the public on the results of 4 & 5	No attempt to perform this activity (politically unacceptable to Govt.)

Figure 7: Activity-by-Activity Comparison of Ideal System With Actual System:

Stage 6: Analysing Feasible and Desirable Change

Obviously, all parties affected by change have not been consulted in this example (I'd probably be sacked for subversion!). However, taking system (7), above, the following observations can be made:

1. Mechanisms to determine what factors make driving more attractive than public transport are in place, but this assessment is not informed by feedback from assessments of measures to limit use of private cars. *Action = provide that feedback.*
2. This activity is performed reasonably well. *No action required.*
3. Actions are being taken, but not consistently in all areas of the country (e.g. wheel clamping used in London, but not Coventry; private contractors used in some areas of the country, but impact differs in different areas). Without clear assessment of what factors are most important, how can we measure if these actions are effective? *Action = provide the feedback mechanisms needed for 1, above.*
4. Measurements are being taken, but numbers and impact are not being fed back to earlier activities. *Action = feed-back results of measurements.*

5. & 6. Not performed and unlikely to be acceptable on a national level. *Action = manipulate public opinion via pressure groups.*

However, an understanding of what change is required is not sufficient. **We need a plan of action** and for this we need to prioritise our recommendations. For system 7, in the example above, the priorities would possibly be:

Priorities:

1. We need to assess determining factors, before we can do anything else therefore Action = provide mechanisms needed for 1, above.
2. Effectiveness depends upon determining what action can be taken, therefore Action = provide mechanisms needed for 2, above.
3. The effectiveness of various measures needs to be assessed nationally, to inform transport policy; Action = lobby Govt. to collect and distribute data on the effectiveness of various measures.
4. Having determined what actions are effective, actions must be changed to match those needed. Action = change actions of local govt. to match those indicated by feedback to be effective.
5. Once effective actions are in place, measurements of their effectiveness are useful. Action = further feed-back of results of effectiveness of measures taken.

Health Warning:

This is just a short example, based upon **guesswork** of what the Government aims and current mechanisms are. It is not intended to be complete, only to gives examples of how to use the tools of SSM. However, it should give you an idea of how SSM analysis can be used and how the objectives of your client should tie up with the recommendations for change.

5. Additional Tools For Use In Various Stages of Analysis

This section describes some methods which may be of use in the stages of SSM analysis. They are not tools which have been used as part of the "official" Soft Systems Methodology, but they are tools which the author of this note has found to be of practical use when applying SSM.

(a) Problem Diagrams

This tool is not part of Checkland's SSM, but was derived by Professor Galliers, in his work with students at Curtain University, Western Australia. A problem diagram can be a useful tool for structuring ideas, once a reasonably full set of rich pictures has been obtained. The problem diagram attempts to investigate problem causality; very often understanding the root cause of problems can stimulate discussion on system transformations to solve them. An example is given in figure 8.

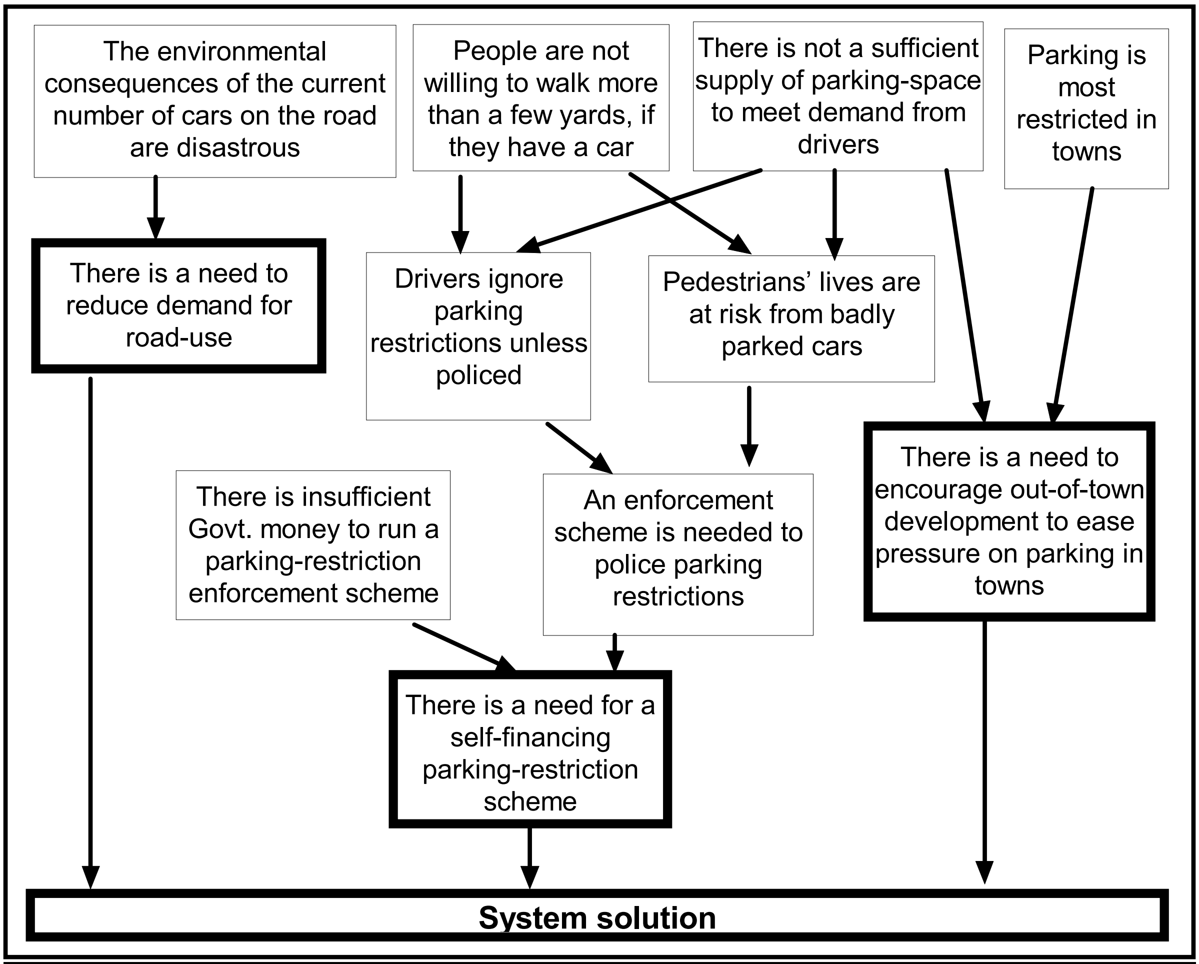


Figure 8: A Problem Diagram For The Traffic Warden Scheme

A **problem diagram** helps in defining some of the system "perspectives": what are the reasons for the new system's existence and what are the objectives that the new system must fulfil? Look for **cause and effect**: what are the root causes of perceived problems? (There is no point in trying to solve a problem which is the consequence of another factor, unless you can affect that factor itself). This can be a great help in defining input-output transformations. It is also helpful to define the system boundary, again, on the problem diagram: which problems are within the scope of the system and which are outside the scope of this system? (This prevents the analyst from trying to solve problems over which they can have no influence!).

(b) Activity-Linkage Diagrams

This tool is useful as a way of attempting to integrate conceptual maps. The main "system" transformations are collected on one drawing and connections are drawn between them, which may represent data-flows between the processes, or may represent a control-link (i.e. one transformation-activity must be performed before another). This is a useful way of analysing if transformation processes conflict with each other, or are mutually incompatible. An example is given in figure 9.

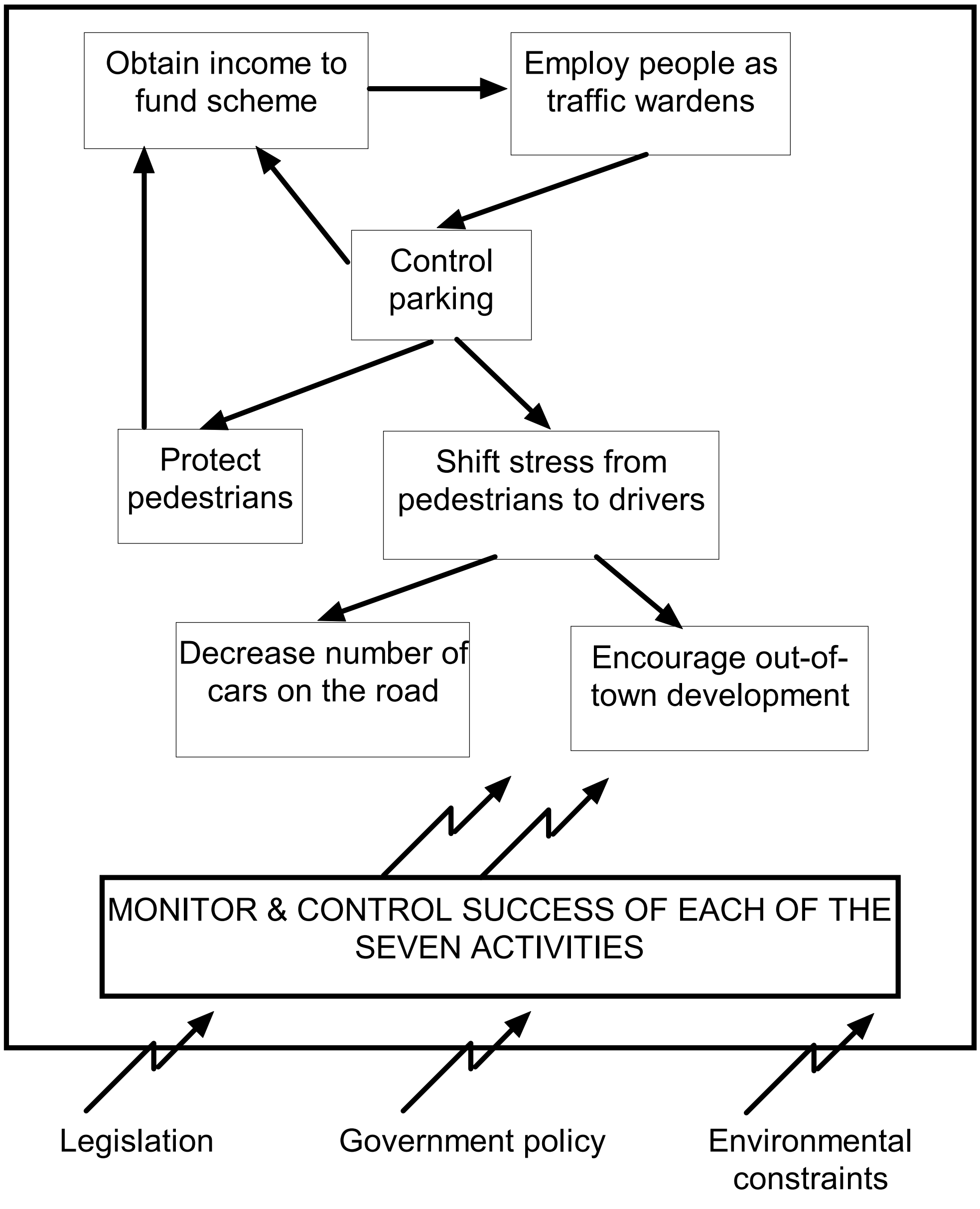


Figure 9: An Activity Diagram For The Traffic Warden Scheme

An **activity diagram** can be used at the *end* of the SSM modelling process, to bring together the system solutions that have been derived through the use of conceptual modelling tools. What are the consequences of implementing one system perspective, on the other system perspectives? How can the various system perspectives be merged into a holistic set of requirements for organisational change? Where are there conflicts between different system perspectives?

(c) Activity Grids

This device was derived by the author, as a structured mechanism for comparing activities in the conceptual model with the real world. A grid is drawn, with the activities listed on the left-hand-side and comments as to the effectiveness of each and its links with other activities on the right-hand-side. An example of this is given in figure 7 (in the worked example, above). The important factor here is to be as detailed as possible in your analysis of to what extent and how effectively each activity takes place and also of how effective the linkages between activities are. This enables you to list the required changes to each "system" needed to move towards the "ideal" system definition.

6. Health Warning: The Final Word

This teaching note has attempted to explain how to use SSM as a method of analysis. It is not a recipe for success and it is not intended to represent the philosophy of SSM; it is intended as a set of notes to enable you to understand the mechanisms of SSM.

For further information on the philosophy and intention of SSM, you are recommended to read the book which started it all:

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

It is also a good idea if you reflect on your own role in the analysis: to what extent have you acted as an interpreter of the aims and objectives of your client or other organisational actors and to what extent have you simply told people what is good for them? A good paper to read (certainly for your project, if not for the first assessment!) is:

Schein, E. H. (1990) "A General Philosophy of Helping: Process Consultation", Sloan Management Review, Spring 1990