Agile SD: fast, effective, reliable.

Kim Warren

Strategy Dynamics Ltd
Two Farthings,Aylesbury Rd
Princes Risborough
BUCKS HP27 0JS, U.K.
+44 1844 274061
kim@strategydynamics.com

> "*When faced with a new problem, I start by identifying the stocks and how they are changing. I do not try to develop causal loop diagrams, though I know this is popular.*" Forrester, 2013

ABSTRACT. *Recent concern about the progress and impact of system dynamics calls into question the means by which the method is deployed. Books, courses and published cases suggest we start by defining how the issue of concern is changing over time, then build qualitative causal-loop diagrams with stakeholders. The resulting shared mental model is taken to both encompass the scope of the issue and represent well the causal mechanisms involved. Stock-and-flow structures are then added to the model and data is sought with which to populate and formulate those structures, so as to create a working mathematical model. The process is difficult, time-consuming and unreliable; risking serious flaws and omissions, and producing different models for similar cases. The science of the method suggests a simpler process, which moves directly from the performance behaviour to a quantified mapping of how stocks and flows are changing. From there, interdependencies are traced – again with quantified support – and significant feedback mechanisms identified. Experience to date suggests that models are easier and faster to build, perhaps cutting the time and effort involved by as much as an order of magnitude, and the method builds in quality from the start. Valuable insights also emerge throughout the process – reminiscent of the "agile" approach which now dominates the field of software development. Early experience suggests the approach may merit more widespread testing to confirm these benefits. The method is also consistent with a complementary approach, common amongst leading practitioners, of leveraging proven structures repeatedly across similar cases.*

KEYWORDS: Model development, methodology, agile method, waterfall method, system dynamics.

Concern has been building over recent years as to the real-world impact of system dynamics and its progress in winning recognition and adoption by those with authority over significant policies in a wide variety of domains – environment, health, economics, business, security and so on. (Forrester, 2007;

Homer, 2013. This, it is suggested, is due to an absence of high quality work, together with a substantial stream of inadequate modelling, as evidenced in particular by material presented at the International System Dynamics Society's annual conference. However, it is equally possible that high quality work is, in fact, being done but since the skilled professionals producing it have neither the channels nor the incentive to publicise great work, only a small fraction of such work is actually visible. Nevertheless, discussions with other professionals in the field suggests that "selling system dynamics" is hard, and even when initially successful, scepticism, disillusion and rejection may result.

Other methods appear to have avoided this reaction from those influential users, and thus gained rapid and widespread adoption. This applies both to relatively simple methods, such as balanced score-card, and to more sophisticated approaches, such as value-based management, systems engineering, 6 Sigma, and business process engineering, each of which has swept their potential domains of application in no more than 1-2 decades. Why, then, has system dynamics not been equally welcome, adopted and impactful?

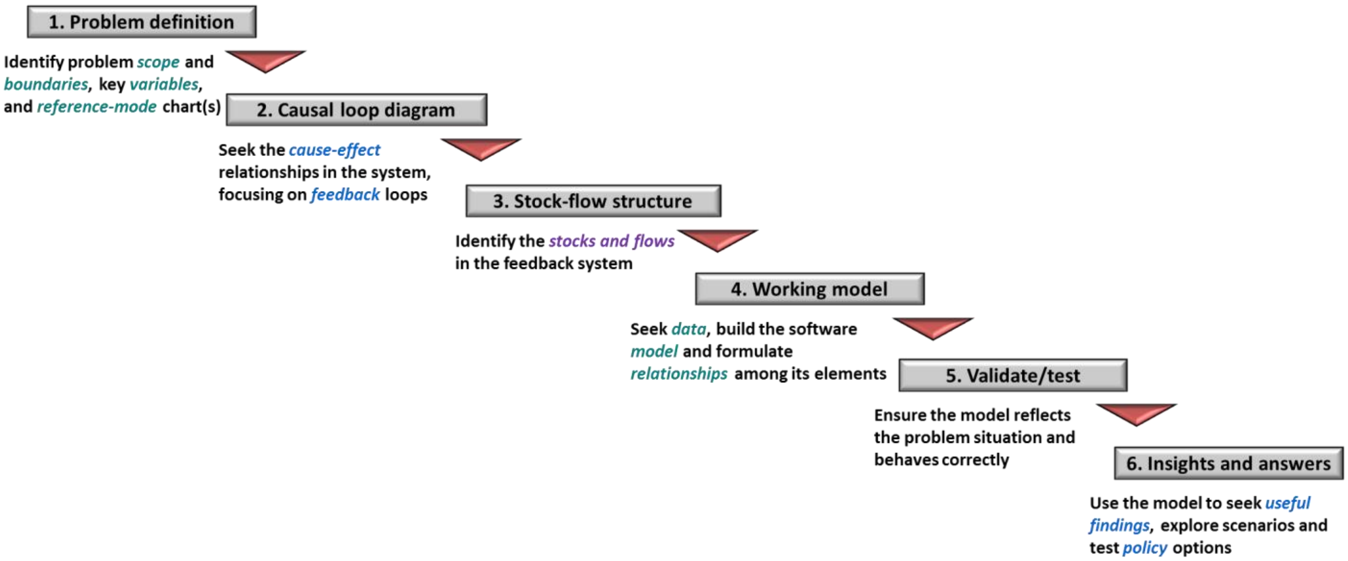
One might reasonably expect a method to be adopted if it meets three criteria – delivering clear and demonstrable benefits, with reasonable effort and cost, and doing so reliably, where "reliable" implies that the method can be deployed with confidence in similar cases, with similar impact. The low recognition, slow adoption and limited impact of system dynamics raise doubts about the method's performance on all three of these criteria.

Mature professional disciplines typically feature some best-practice by which their methods are implemented – standard procedures to ensure reliable delivery of valuable benefits, at reasonable cost. Professional societies typically publish such procedures (see for example the Supply Chain Council's SCOR frameworks, the Balanced Scorecard Institute's Nine Steps to Success, and INCOSE's Systems Engineering Handbook).

Although no such closely defined procedure exists for how system dynamics models should be developed, guidance offered by key sources exhibit considerable similarity (Lane, 1994; Vennix, 1996; Sterman, 2000, p. 86; Maani and Cavana, 2000; Morecroft, 2007; Pruyt, 2013). Although alternatives exist, such as the modular approach (Wolstenholme, 1994), most conference presentations and published articles suggest that the guidelines from the former sources are indeed widely followed. An important exception, however, is that many professionals appear to follow somewhat different practices, often adapting previously proven structures to similar cases. This practice offer important insights regarding the practice and adoption of system dynamics, discussed later.

The apparently common method (summarised in Figure 1) pays attention first to identifying the scope of the problem and identifying the outcome of concern – plotting a chart of how that outcome has changed over time, and might change in future. This is a quantified task, both as regards the value of the outcome indicator, and the time-scale over which change occurs, and it results in the "reference mode" against which the resulting model is assessed, scenarios explored and policy options tested. There may be more than one outcome of concern – the reduction in some harmful factor *and* the cost of that effort, for example – so time-charts of more than one item may be needed.

Figure 1: Simplified diagram of a common approach to developing system dynamics models



In step 2, people involved in the situation are asked for their views of what other factors may be causing that outcome, and for the causal relationships they believe may exist between those factors. Particular emphasis is placed on identifying feedback loops in the causal structure. Some collective process is used to negotiate and combine these views into a single causal-loop diagram (CLD), taken to represent stakeholders' shared mental model of how the system driving the outcome of concern is structured.

In step 3, any accumulating stocks that may exist in the diagram are identified, and a modified diagram produced in which those are made explicit. Some practitioners identify stocks and flows in the initial diagramming, although this adds to the burden of explaining the method to participants. In some cases, steps 2 and 3 may be combined, with facilitators attempting to develop causal diagrams that include stocks and flows from the start, an alternative that brings some benefits, but also some costs (Lane DC, 2008; Fisher, 2010).

In either case, the feedback diagram with stocks and flows provides the platform for step 4, where data is sought and arithmetical relationships specified so that a working quantified model can be produced. Although simply stated, this step is a large and technically demanding part of the model development process.

The model is then validated and tested in various ways, including confirmation that it does indeed explain the reference-mode behaviour of the outcome indicator(s), after which the model can be used to test alternative scenarios and policy options. Although Figure 1 suggests that model-development is a one-pass, linear process, Sterman (2000) emphasises that modelling should be an iterative process, with lessons from one phase of the process leading to revisions to the outcomes from earlier steps. However, it is not evident from published work whether this iteration is commonly carried out.

Problems with the standard process

Although the process summarised in Figure 1 may appear to conform with the field's accepted principles, a careful assessment suggests it may not be easy to perform, effective or reliable. This is due to its neglect of important issues that are well known and accepted in the field, its attempt to "boil the ocean" by mapping the entire problem space from the start, its reliance on purely qualitative methods in the early stages, and the staged phases of development it implies (whether or not iteration is added to the process).

Step 1 – plotting a quantified time-chart for the outcome(s) of concern – is usually straightforward. However, attention is rarely paid to whether that item is a stock (fish population, for example) or is instead an indicator whose value *depends on* one or more stocks (the fishing catch-rate). Skilled SD facilitators, of course, know exactly which is the case, but newcomers to the method will not, and the perceived difficulty of explaining this distinction may lead to the participants being spared this conceptual complication.

Step 2, is more problematic. First, having quantified the performance indicator's time-path, the search for causal relationships proceeds in an entirely qualitative manner. Stakeholders proffer their view that factor B depends on factor A, whether or not any evidence exists to support that assertion. Nevertheless, their status as actors in the system is taken to justify including that relationship. Yet it is axiomatic in system dynamics that human cognition is not capable of understanding how feedback systems work (Sterman, 1989 and 1994; Paich and Sterman 1993; Moxnes, 2000). Furthermore, human perception is notoriously subject to selection and bias (Kahneman , Slovic and Tversky,1982 ; Haselton, Nettle and Andrews, 2005). It is therefore highly implausible that the sum of individual mental models will describe in any reliable manner how any system actually functions.

Qualitative consultation raises the further risk that social and power relationships amongst participants influence the content and structure of the CLD and resulting model. Important factors or links may be missed, simply because those with that knowledge are not consulted, because they hesitate to offer their views, or because those views offered are ignored. Accepted wisdom can also get built into the model, even though that wisdom is itself dysfunctional.

The second problem is that step 2 typically avoids identifying stocks and flows, even though the accumulating behaviour of stocks is fundamental to any system's behaviour. (*The alternative procedure of causal-mapping with explicit stocks and flows from the start appears to be rare*). But it is again axiomatic that people cannot reliably estimate the behaviour of stock-flow structures (Cronin, Gonzalez and Sterman, 2009; Cronin and Gonzalez, 2007; Sterman, 2010) and since at least one accumulating stock *must* exist in any feedback loop, every such loop must include *at least* one causal mechanism that people do not understand. The omission of stocks from the CLD process is especially dangerous where those elements are made up of aging chains (which rarely appear in CLDs in any case) since these exacerbate the stock-item's influence on system behaviour. Articles published in SDR during the past 10 years include virtually no cases of CLDs in which aging-chain states are distinguished, although such chains may appear later in stock-flow developments of those CLDs. This is a further example of the recommended and widely practised process for model development neglecting long-standing and fundamental knowledge in the field (Morecroft, 1982; Richardson, 1986).

Thirdly, step 2 depends on the reliability of the relationships between feedback loops and their behaviour – reinforcing feedback produces accelerating growth or decline, and balancing feedback produces goal-seeking behaviour. However, it is entirely possible that feedback loops of either type exist *without* generating such behaviour, due to the existence of other causal mechanisms. It is also possible for the behaviour ascribed to feedback loop to arise from other mechanisms entirely. Accelerating growth, for example, can result when a factor changing in a linear manner passes a threshold, as for example where a product's performance or cost make it attractive to potential customers, or when rising work stress triggers staff turnover. Feedback *may* add to these effects, but they also operate alone.

Problem owners can therefore neither intuitively estimate outcome behaviour from inspecting the feedback structure of a system, nor intuitively estimate the system's feedback structure from inspecting the outcome behaviour, problems exacerbated by the absence of data from step 2.

Fourthly, the justification for step 2 relies on the belief that the "mental database" of participants far exceeds the written or numerical information available. Yet the sheer quantity of numerical information available in any situation has long exceeded than which any human being can carry in their head. Even if critical information *does* exist in human memory, the task of extracting it is challenging – the first three issues in step 2 imply a high likelihood that we will not seek all relevant information, and may obtain much that is irrelevant, and even if we seek the correct information, it will be subject to bias and misperception.

If the argument is, rather, that the mental database contains information of greater *importance*, then the same issues arise again. Just because an actor *thinks* some factor is important, and has great knowledge about it, their inability to comprehend how the system works may mean they are wrong, and that they will miss other important factors. In practice, any actor can have reliable information only on that part of the system in which they personally are active. We should not care, for example, what any person believes about how a company's customers behave; the only reliable information will come from customers themselves (and even that may not be reliable!) or from data about their observed behaviour.

An over-arching problem is that step 2 seeks to identify and encompass the whole of the problem space of concern. Since the entire step is carried out with no data, it is not clear how we can know whether the activities in step 2 have indeed captured the whole problem space or alternatively strayed beyond its boundaries. A focus on participants' views on feedback – or "closing the loop" – also risks neglecting important exogenous factors. The common advice on system dynamics modelling is to focus on the endogenous causes of system behaviour, but few issues of concern are *entirely* immune to forces over which participants in the system have no significant influence.

Having agreed the shared mental model of the situation, step 3 of the process should not be problematic. Accumulating stocks are found in common categories – people or other populations, cash, physical assets, and so on. Nevertheless, novices generally struggle with this step, both incorrectly identifying items as stocks, and missing items that *are* stocks. Failure is especially likely where aging chains of stocks are involved (even assuming they were identified in step 2). It seems, then, that only experienced system dynamics practitioners can reliably identify stocks in qualitative feedback diagrams.

Step 4 – building a quantified working model of a problem – follows established model-building procedures, but must be done by those same experienced system dynamics practitioners who can facilitate steps 1-3. Typically, it is only at this stage that significant numerical information is looked for, and since the wide audience previously consulted has already signed up to the feedback diagram, that structure determines the entities for which data is sought, unless of course the modeller is sufficiently experienced that they identify important missing items, take those back to the audience, and argue for their inclusion in the model. There is thus a substantial risk that data about important factors is not looked for, because it was not identified in the CLD, and that data about insignificant factors is forced into the model, regardless of its relevance.

Step 5 – validation and testing – also follows accepted procedures, on which there is an extensive literature (Forrester and Senge, 1980; Graham, 1980; Barlas, 1989; Coyle and Exelby; 2000; Peterson and Eberlein, 1994; Sterman, 2000, Chapter 21; Schwaninger and Groesser, 2009; Yücel and Barlas, 2011). If the common process in Figure 1 were to be followed in practice, however, much effort will have been expended, much time will have elapsed, and great commitment to the model structure will have been built before this vital task is carried out. Fundamental problems could easily survive from the very start of the process, making it necessary to go way back to the beginning and revise all of the prior work! Skilled modelers no doubt avoid this danger by keeping an eye on their model's structural validity throughout steps 2 and 3, and on its mathematical validity during step 4.

Step 6 is where the simulation model is actually used to answer the questions identified in step 1. For a model to make a difference to any organisation's performance, someone must do something additional to or different from what would otherwise have been done – and such action must have some scale and occur at some points in time. Since these are quantitative features, they must be informed by quantitative findings. So, whilst qualitative insight might arise as early as step 2 in the process, participants cannot confidently act on anything until steps 3 to 5 are completed.

Adverse consequences for modelling projects and for the field

Although much insight may be gained, and successful simulation models developed from the commonly accepted process, it is reasonable to fear that the problems outlined above have caused serious adverse consequences, both for particular projects, and for the wider adoption and impact of the method.

A shared mental model is widely regarded as critical to developing a good system dynamics model, so the topic has received substantial attention. Evaluation of individual, group and method outcomes has typically fallen into three categories: participant satisfaction and acceptance (McCartt and Rohrbaugh, 1995; Vennix, Scheper W and Willems; 1993; Huz, Andersen, Richardson and Boothroyd,1997), changes in participants' and group thinking (Franco and Rouwette, 2011), and improvements in participants' capability (Scot, Cavana and Cameron, 2014).

However, when viewed from the perspective of *impact*, these questions are not of primary concern. Two questions are intriguingly absent from the criteria for assessing the value of building shared mental models. First, is the shared mental model actually valid in any case, or does it contain significant objective omissions or errors? Secondly, what did anyone *do* as a result of the process (spend more or less money, commit more or less effort and so on), with what significant improvement in outcomes? Since a most important consideration that might lead to widespread adoption of a method is whether its use made a worthwhile difference, this latter omission is of particular concern.

Also absent from this literature is the question as to whether any alternative process might lead to the same, or better, changes to participants' understanding of the situation in which they are involved, to their decisions about what to do, and to the resulting performance changes.

Not only is it unclear what value arises from step 2 in the process, it is also time-consuming and costly to perform. It takes a long time to agree and engage the wide audience whose shared mental model is to be developed, and several events may be needed to develop and refine that model. A separate effort must then follow to specify and build the working model, including lengthy efforts to find required data. Consequently, long periods elapse between launching the effort and obtaining actionable answers.

The absence of data in the process or any objective validity-checking make it likely that CLDs and models will omit critical items (especially stocks) and causal relationships (especially multi-stock-flow structures), and include irrelevant or mis-stated items and links. The field has long shared a joke at the expense of economists about an econometric model of milk production that included no cows, but published SD models include similarly surprising omissions. A professionally built model of an insurance company, for example, was found to have no stock of policy-holders, and a model of the prosperity of a fishing fleet did not include the price of fish.

A further consequence of such omissions is that models cannot be transferred to other situations, no matter how similar they may seem. Non-transferability of models is an unavoidable consequence of the belief that the CLD built for any case must reflect the shared mental models of those who were consulted. That same CLD cannot, therefore, provide an acceptable start-point for any subsequent project on the same or similar topic with any other group. This belief stands in stark contrast to other professional disciplines, which typically feature repeatable solutions – a method that has been devised to apply the field's methods to a common class of problem. That procedure is then tested, refined and documented, from which point it can be deployed (with suitable adjustments) to many similar cases. The tendency for SD projects to "model the problem, not the system", if it were strictly observed, would leave the field unable to offer such repeatable solutions. As Forrester (2013) remarks "*Seldom, if ever, should a person model the specific situation of interest but, instead, should model the family of systems to which the specific one belongs*", or as Homer (2013) puts it, that our method should "… *be able to project a comprehensible 'sameness', as other modeling disciplines have managed to do*".

Together, the problems with the commonly accepted process seem to explain why SD projects might be costly, time-consuming and of uncertain value. And if that is the case, then persuading any potential client to trial a first project or to repeat the effort for other challenges will be difficult. We thus have a plausible explanation for the slow adoption and limited impact of the method.

The problems also cause difficulties in developing professional capacity in the field. The complexity and uncertainty in the method mean that there is little chance of any novice developing from first principles a good working model of even a simple system. Guarding against the risks in the process relies entirely on the expertise and stature of the facilitators, making it challenging for inexperienced practitioners to achieve successful projects. And since the real value only emerges at the end of a long and costly process, it is also difficult for young professionals to win projects in the first place or demonstrate value as the project progresses.

An agile alternative

The process suggested in Figure 1 bears a remarkable similarity to the “waterfall” model which dominated the field of software development until the 1990s. That process too starts with identifying the full scope of the desired solution, then attempts to define the entire architecture of the application to be developed, before the whole solution is coded. The software is then tested and debugged, before being installed for users to employ.

Like many SD projects, software solutions produced with the waterfall development process were also widely viewed as taking too long, costing too much, and delivering uncertain value (Fowler, 2001). It has since been superseded in many cases by incremental and adaptive “agile” methods, in which user engagement and satisfaction are paramount, and achieved by continuous delivery of working software. Other principles include close collaboration with knowledgeable individuals, rather than set-piece reviews with large groups of stakeholders, welcoming changing requirements driven by learning as the software evolves, simplicity (maximising the amount of work not done), and technical excellence – building in quality from the start, rather than testing and debugging at the end (Abrahamsson, Salo, Ronkainen and Warsta, 2002).

Can some of the features and benefits of this agile alternative in IT development be mirrored in the building of system dynamics models? Such a process would still require technical excellence, but that might be ensured by observing the rigorous science on which the method relies, consisting of four simple principles:

1. We seek to explain, anticipate and improve how outcomes of concern change over time.

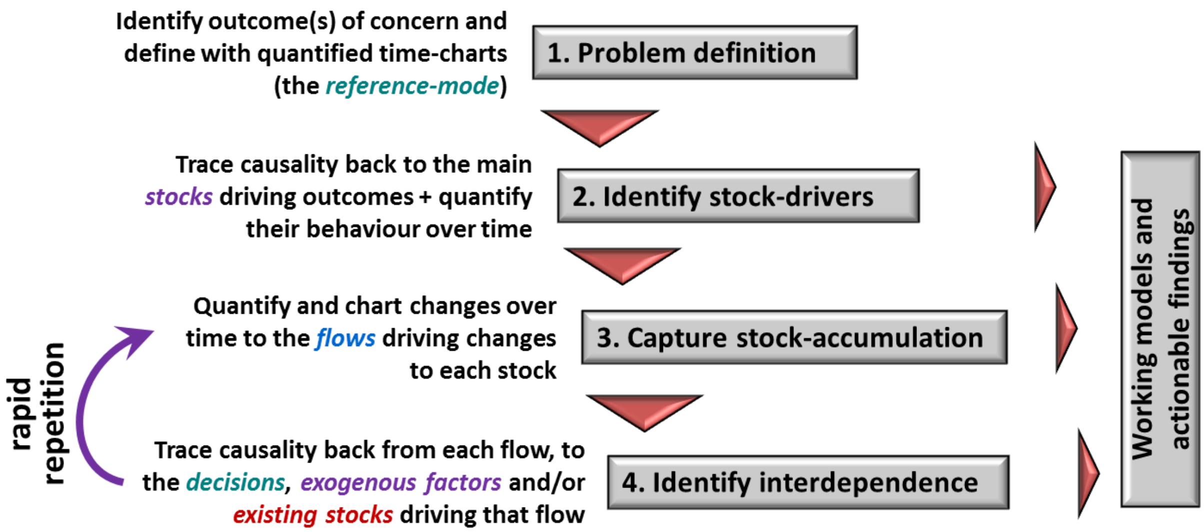
2. Those outcomes arise at each point of time directly from the quantity of accumulating stocks (*except where the outcome of concern is itself already a stock*). The behaviour over time of such outcomes therefore depend directly on the behaviour over time of the stocks on which they depend (*plus any changes caused by exogenous factors or actors in the system*).
3. The quantity of any accumulating stock depends, exclusively and absolutely, on all previous values of their associated flow rates.
4. Those flow rates depend at each point of time on *existing* stock quantities, decisions of human actors, and/or exogenous factors.

Items 3 and 4 together create interdependence, including feedback, but that feedback is a *consequence* of the underlying principles, not in itself one of those principles.

The logical progression of these principles can inform an agile process for SD modelling (Figure 2). In the first iteration, each step is taken directly with the problem owners or concerned stakeholders in a matter of hours. A diagram is built out from the factor(s) of concern in which every item and every causal relationship between those items, is supported by quantified time-charts, even if values have to be estimated. Subsequent iterations can happen with individuals or small groups with particular knowledge of the parts of the model under scrutiny. Every step is also supported by a working model, to check that the emerging causal analysis is valid and realistic. The method should be seen as a radical adaptation of, rather than replacement for, other group model-building processes (Vennix, 1996).

It is possible that practitioners who have long recognised the problems with the commonly recommended approach have adopted similar procedures, although the extent to which such a procedure is used is unknown. Even if it is used and helpful, it needs to be documented and subjected to further testing.

Figure 2: An agile model development process



In order to demonstrate the procedure, the following paragraphs develop an example showing the diagnosis of a problem situation facing a mid-size IT-support company. The case is extremely simple, and is chosen to most clearly demonstrates the principles of the method.

Case example: A mid-scale IT-support company

The company provides IT support to mid-scale businesses, such as small retail groups, accounting and law firms, and small construction and transport companies. It supports clients' needs for computer hardware, software and communications, and employs young, skilled technical staff to do so. After some successful years, the company embarked on a growth effort, which initially was successful, but some two years later was facing high rates of client problems, and the loss of clients.

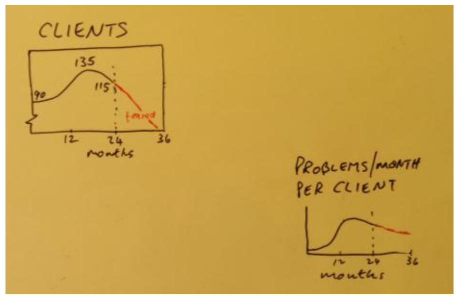
Step 1 is exactly the same as in the standard process – draw a time-chart(s) of the issue(s) of concern, including relevant history and desired future. The time-chart <u>must</u> have a scale on it and the chart's line must reflect at least an estimate of how the performance indicator (call it 'A') has actually changed. Clarifying if this chart is for a stock or something that *depends on* a stock must be done at this point, because it determines whether to do step 2 or jump straight to step 3.

Step 1. How performance is changing over time

<u>You say customer problems are increasing and clients numbers are falling, and that this issue has developed over the last 2 years. Can you explain what has happened and sketch a chart of each item</u>? *2 years ago we had about 90 clients, and were doing well, so took on a sales executive to sign up more new*

clients. He did that and we grew to about 135 clients, but about that same time we had reports of rising client problems, peaking at nearly 3 per month for every client. Staff have to fix those problems fast, rather than get on with their normal job of client support. We have since lost some long-standing clients and are now down to 115. We fear it will take time to reduce the rate of problems, and in the meantime clients will continue to be lost. (Figure 3)

Figure 3: The time-charts of concern to the IT-support company



In step 2, the direct explanation for the outcome variable's values is identified, which *must* include one or more stocks (unless the outcome of concern is itself a stock, in which case this step is unnecessary). This is often a simple calculation from 2 or more other items (call those B and C), although sometimes a more complex relationship may be involved. It should be possible to calculate or estimate A from B and C, so time-charts are also added to those causal factors, to prove that the causality is valid. This enquiry is repeated in turn for each of B and C and continues, with values or time-charts for every item identified, until the causal relationships reach one or more of the following items - stocks, decisions or exogenous factors. (*This process does not insist that system behaviour is entirely explained by exogenous relationships. Real-world cases frequently feature important influences arising from outside the reasonable boundaries of the managed system*).

This step takes little time, because items driving performance outcomes are usually identified within just 2-4 causal steps. There can be some uncertainty in this step, but that is resolved by seeking numerical evidence to support the causal relationships involved, by unit-checking, and by developing in parallel a working model that captures the system's behaviour as it is built out. This is shown in figure 5 (after correcting whiteboard estimates), where red chart lines indicate the management's feared future, and some actual data for clients' problem rate. The model is at http://sdl.re/m203, which requires a latest-version browser.

Step 2. How stocks drive performance

<u>What caused the increasing rate of client problems</u>? *It looks like it was the pressure of work on our staff. We log the work they do – installing equipment and software, upgrading systems, training users and so on – and that increased much more than our capacity to do the work. People were working into the night and at weekends. We estimate the workload may have been at its peak 60% more than people could do in a regular work week.* <u>Has anything changed the amount of work each customer needs, like new software releases or equipment needs?</u> *No more than usual.* <u>Did you have problems with unskilled or unproductive staff</u>? *No.* <u>So what happened to staff numbers</u>? *We managed a slow increase for the first half of the period, then numbers fell to about 37 today.* (Figure 4)

Figure 4: Stocks driving performance at the IT-support company

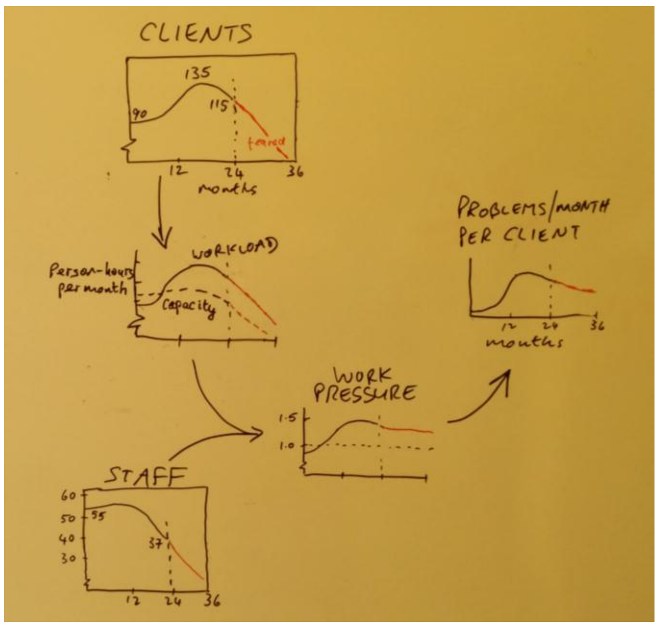
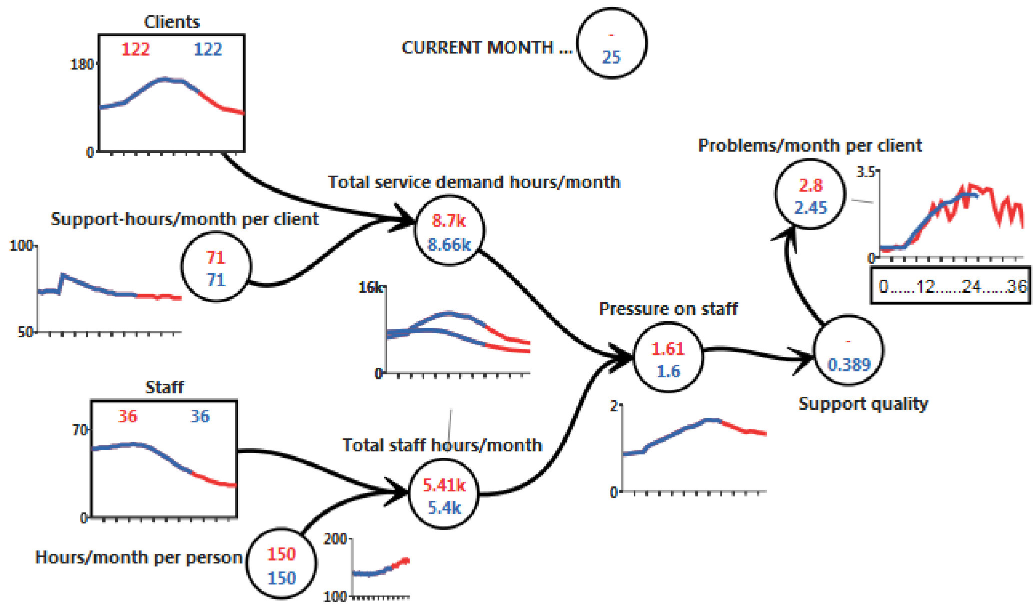


Figure 5: The model of stocks driving performance at the IT-support firm



It is common, of course, that data required to validate these causal pathways from stocks to outcomes is not easily available, or available at all. However, the process can proceed with plausible estimates by participants, and is somewhat self-validating – if A depends on B and C, then data on A and B implies values for C.

Critical to this process is that *both* the diagrammatic picture *and* the working model of the system develop in parallel. Modern SD software is sufficiently user-friendly that there is no reason to develop the diagrammatic representation first, and follow that with a separate, lengthy procedure to put that representation into a working model. A first-pass model built alongside the diagram may need to be checked and improved, but stake-holders should be able to see their working model develop in parallel with their discussion of that model on white-boards or other media.

Parallel development of the diagram and model offers important benefits. Since the participants see their own explanation of how the system works develop in front of them, they immediately trust that model. Secondly, that model includes data they recognise for every element in the model – at least estimated – so no abstract or ill-defined items can find their way into the model. Quality is thus built into the model from the start, and maintained throughout the process (Barlas, 2014).

Step 3 explains the behaviour over time of stock items. That behaviour must reflect, and *only* reflect, how the stocks' in- and out-flows have changed over time, so those too are populated with time-charts. If the issue of concern identified in step 1 is itself a stock, the process starts here and misses out step 2. There can be no arithmetical ambiguity whatever in Step 3– the value of each accumulating stock is identical to the cumulative sum of all in- and out-flows to date. There may, however, be uncertainty when multiple flows exist, again resolved by reference to numerical evidence.

The model is again developed in parallel with the diagram to confirm that the behaviour patterns are consistent (Figure 7, after correcting estimates from the diagram; see http://sdl.re/m302).

Step 3. How flows change the quantity of stocks

How fast have you been adding and losing clients? *The sales guy was successful at first, he upped the new client rate from 2 per month to about 7, but he has struggled to keep that up and now we are hardly winning any clients at all. Previously, we rarely lost any clients, but starting about a year ago, that increased to a peak of 8 a few months back. We are losing fewer now.* What about staff hiring and losses? *We kept hiring about 2 people per month as before. We previously lost people occasionally, but in recent months many more have left.* (Figure 6)

Figure 6: Flow-rates changing numbers of clients and staff at the IT-support company

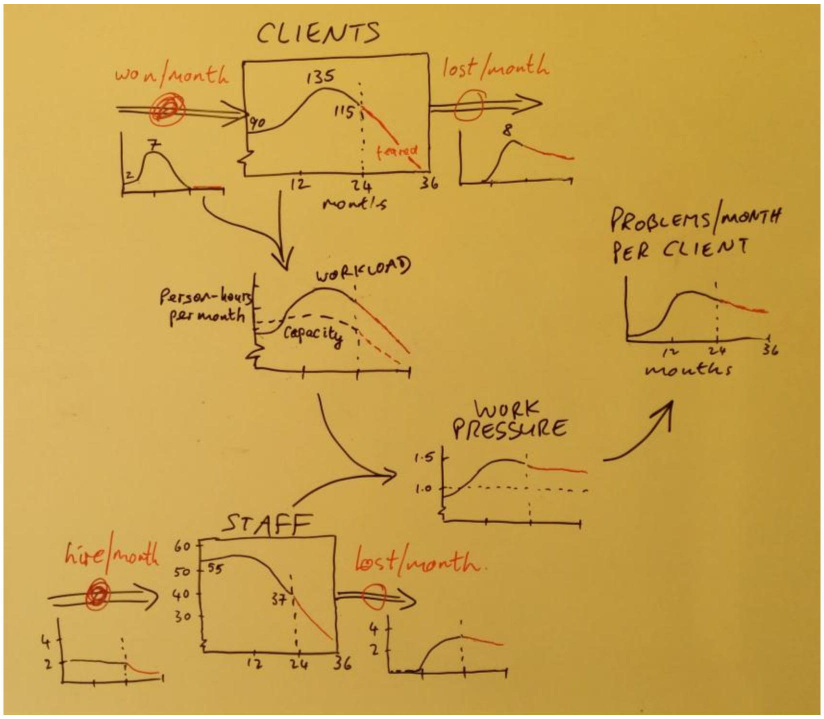
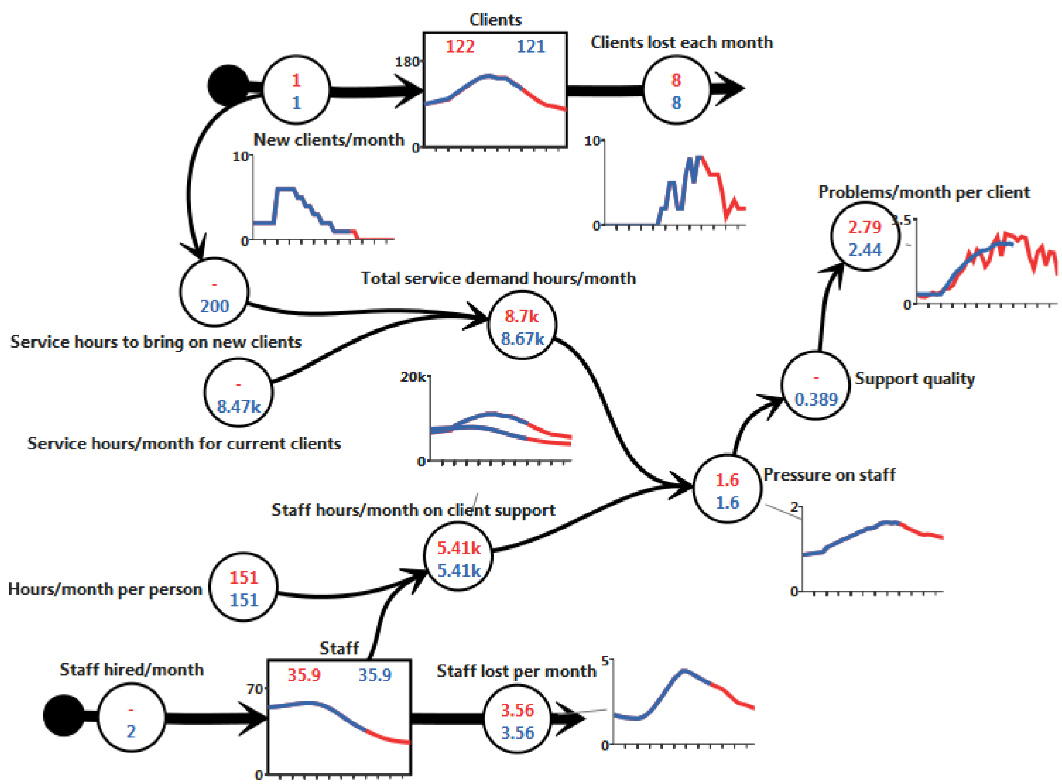


Figure 7: Adding to the model the flow-rates changing the IT-company's clients and staff



By this point, the outcomes of concern are entirely and reliably explained by changes to stock quantities (no decisions or exogenous factors directly affect performance in this case, although that is often found in other cases), and changes to the stocks are explained by the varying flow-rates. The only remaining question is what has caused the flow-rates to change over time. This is answered in step 4 by following the same causal logic of step 2, and once again, every causal chain must originate at one or more stocks, decisions or exogenous factors. And once again, every such chain is validated by at least estimated values and time-charts for all the items along that chain.

Step 4. What causes the flow-rates

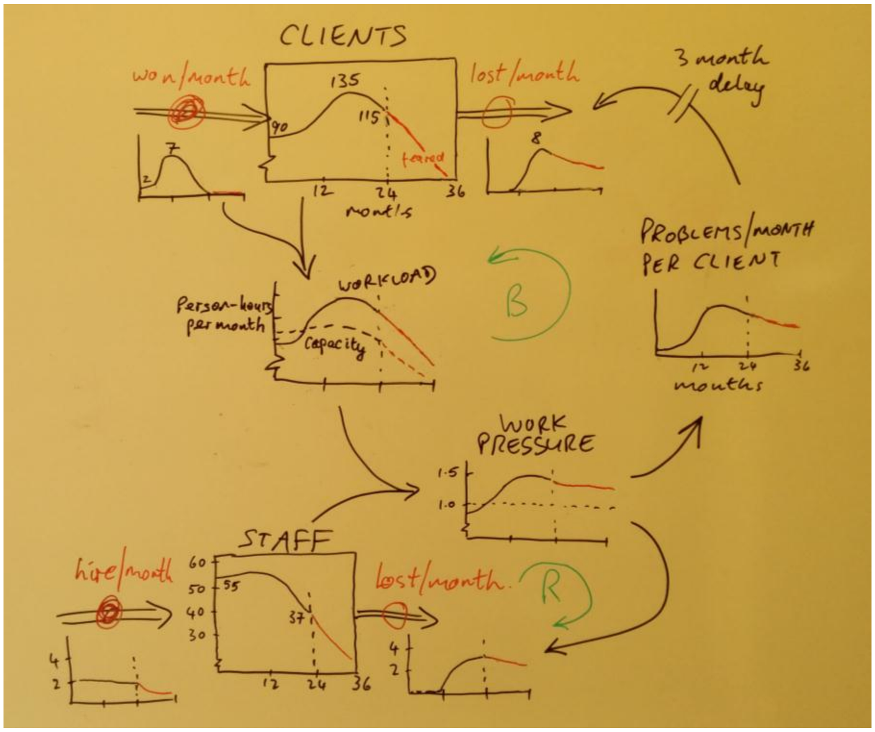
<u>So the growth in clients was simply a result of taking on the sales guy</u>? *Correct – we didn't pick any up from competitors, or so far as we know from recommendations of existing clients.* <u>And why did clients leave at the increasing rate they did</u>? *Well I guess it was inevitable. We were getting calls about problems that we created for them, or delays in fixing things that went wrong. It is a nuisance and takes a while to find another provider, but if you have had a lot of problems over many months, you will probably take the jump.* <u>What about staff hiring and losses</u>? *By the time we realised we had a problem, it was too late to bring in more staff (which takes a while in any case). The faster turnover was inevitable too. People cope with over-work at first, but after a while they have had enough and leave, and it's easy for skilled people to find other good jobs.*

Since any of the causal explanations for each flow-rate may involve any of the existing stock levels, including those of the stock whose flows are themselves being assessed, step 4 identifies interdependencies and feedback (Figure 8). In contrast to the standard process, however, the existence and relevance of such loops arise from the evidence, rather than from participants' speculation.

Step 4 continued. Interdependence and feedback

<u>So the client loss rate reflected poor service, caused by overload arising from having too many clients and too few staff</u>? *Looks like it, but at least we have a bit less pressure, now client numbers have dropped.* <u>… and staff losses were also driven by the high number of clients and too few staff</u>? *I guess so. The problem now, though, is that every person who leaves puts more pressure back on the smaller number of people who are left, which risks driving still more staff away.*

Figure 8: Interdependence and feedback between stocks and flows at the IT-support firm



The interdependence and feedback in step 4 are also added to the model, to confirm the validity of the causal relationships and overall system behaviour. The framework and model are now at a point where decisions to improve performance or solve the problem can be taken, and action-plans developed. The solution and action-plan for the IT-support firm was to drop clients who were causing disproportionate difficulties, by helping them find alternative providers. This quickly killed the work overload, allowing support quality to recover. After a period of additional hiring, the firm was able to start growing once again, although now more carefully. This "preferred" future can be added to the diagram and tested in the model (figure 9, and figure 10, which plays out one solution scenario; see http://sdl.re/m4a03).

Step 4 may also identify additional stocks, not found in step 2 (although that is not the case for the IT-support firm). Those additional stocks may not be involved directly in explaining outcome behaviour, but *are* involved in explaining the flow-rates of those resources that do drive behaviour. A company's product range, for example, may not directly feature in explaining its profits, but be part of an explanation for its customer win-rate.

Figure 9: Sketch of a solution for the IT-support company (blue chart lines, from month 24)

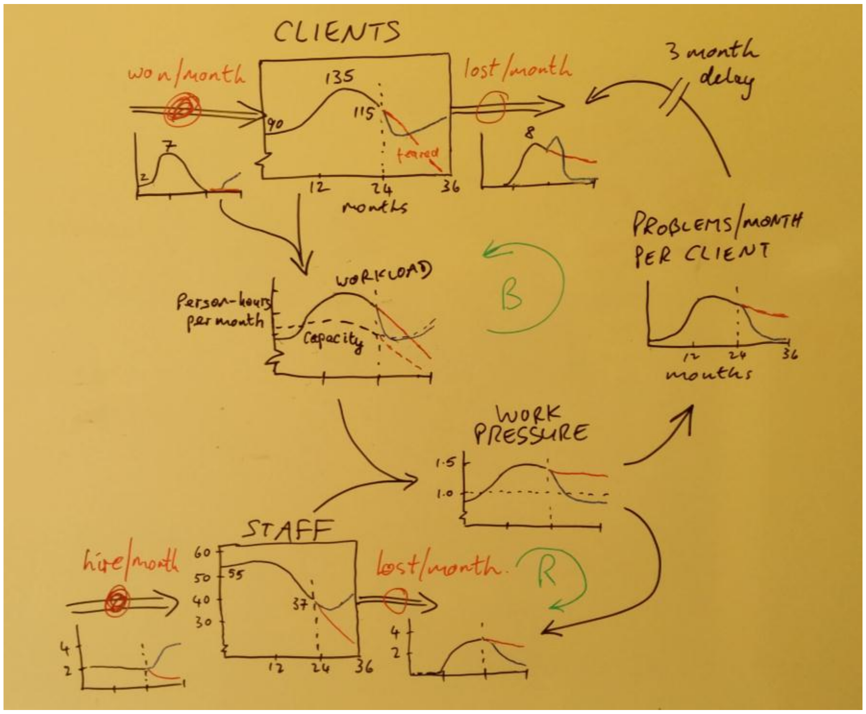
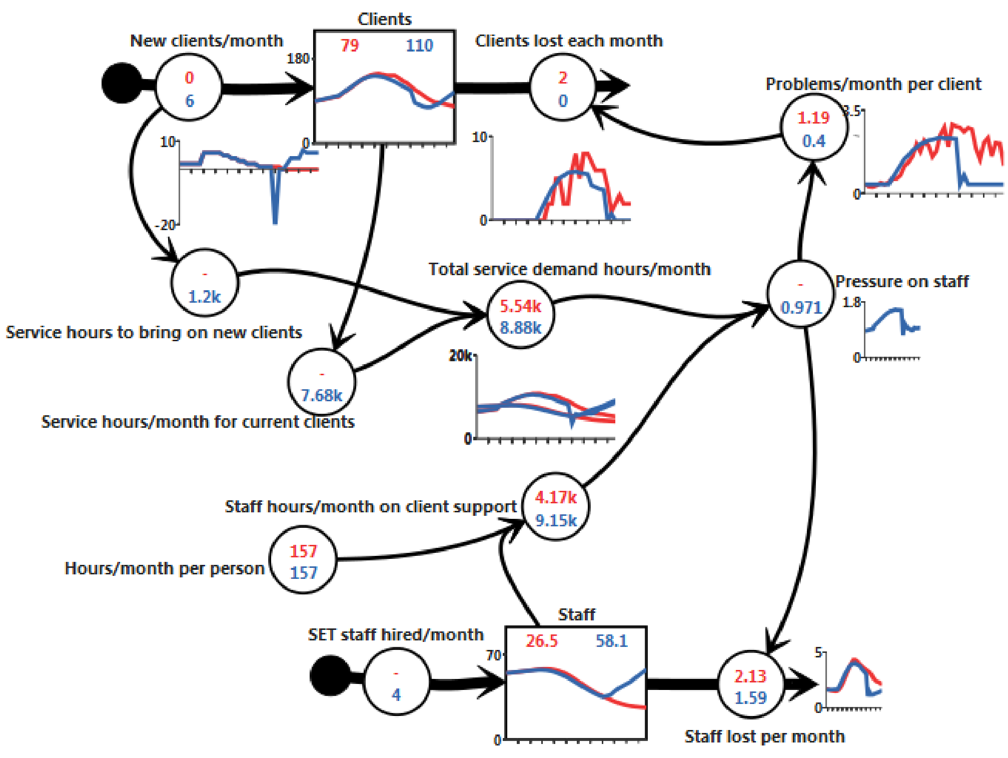


Figure 10: An example of a solution test in the IT-support company model



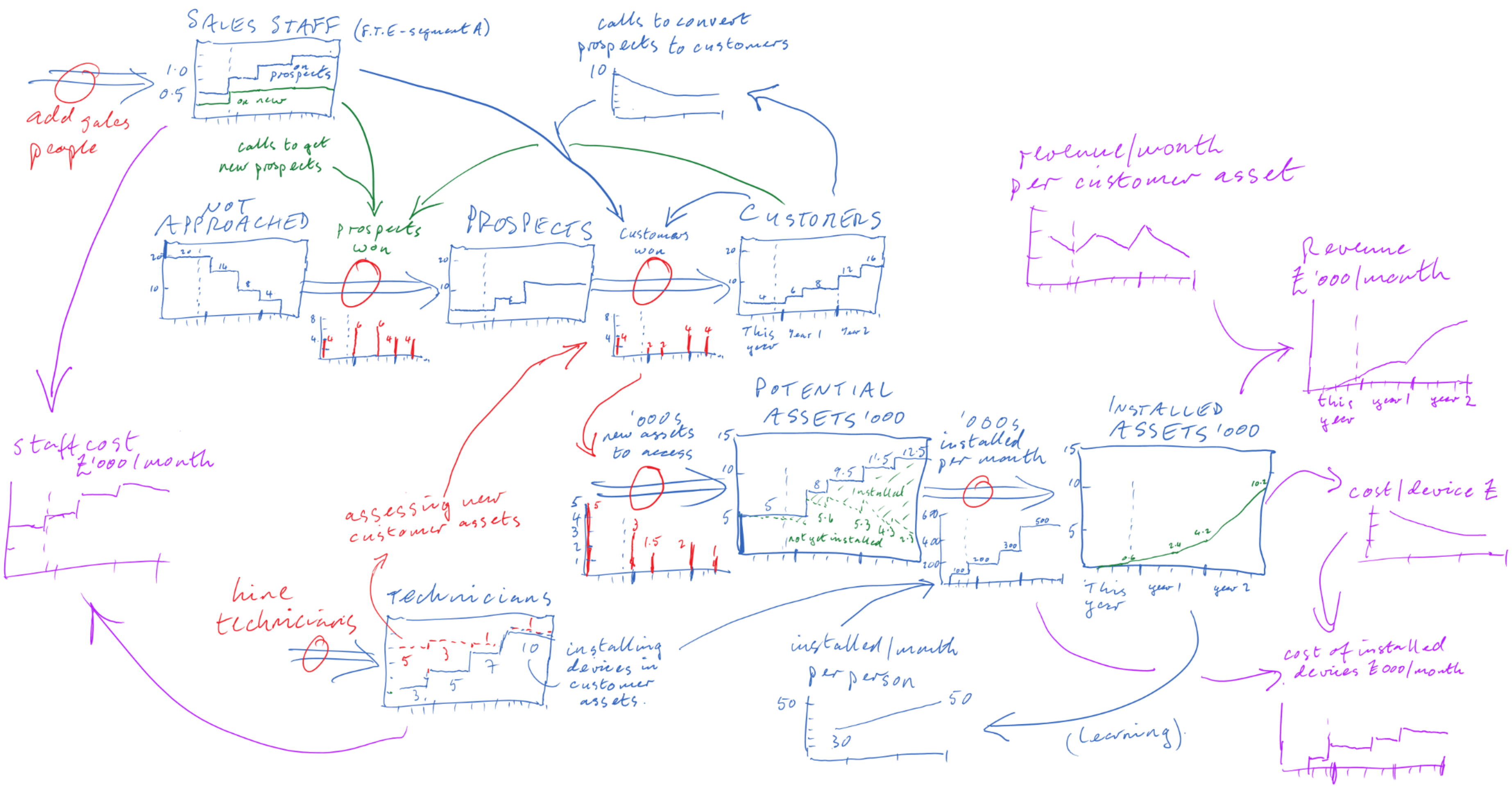
These additional stocks may, in particular, include prior stocks and subsequent stocks in aging chains – explanations for changes to a stock of adult fish, for example, must include the stock of juveniles, or a company's graduate hiring rate will reflect the stock of available candidates. Such additional stocks are readily identified by posing the question "*From where does this flow originate, and is its source important?*" or "*To where does this flow go, and is the destination-stock important?*" The only such additional stock in the IT-support case is the small number of trainee staff, who take a few months to gain experience and become fully productive (see the model at http://sdl.re/m602).

A more substantial case of sequential stocks is shown in the board diagram replicated in Figure 11, which concerns a high-tech hardware firm, 9 months through its current financial year, and wanting to plan marketing, sales, and staffing priorities for the next two years. Its revenue comes from the installed base of energy-cost-saving devices in clients' equipment assets. A fraction of "full-time-equivalent" (FTE) ***sales staff*** try to capture supermarket chain customers, who are the main market for its products (other segments include data centres and certain industrial users). Each potential customer moves from a stock of ***not yet approached*** to become ***prospects*** and then contracted ***customers***. Feedback arises because customers already won reduce the number of ***calls to convert prospects*** to customers, balanced by declining numbers and value of remaining customers not yet approached.

Some of the company's ***technicians*** then visit each new customer's sites to assess which of their assets are suitable for the device. Each new customer thus brings with them a stock of ***potential assets*** which are converted into ***installed assets*** by further technician visits – the faster devices are installed, the more quickly the stock of potential assets ***not yet installed*** is depleted (green dashed line).

Financial values are omitted for commercial reasons, but the company makes ***revenue*** (purple section at right) from a percentage of the customer's energy-cost saving. Assets offering the largest saving are installed first, so ***revenue per client asset*** is initially high for each new customer, then falls as lower value assets are converted. Feedback arises because technicians and the processes they follow become more efficient, so the number of devices ***installed/month per person*** falls. The ***cost per device*** also falls due to experience-curve and economies of scale. Profit (not shown) can then be calculated by deducting the ***cost of installed devices*** and ***staff costs*** from the company's revenue.

Figure 11: Stocks and flows of clients and equipment assets at a high-tech firm.



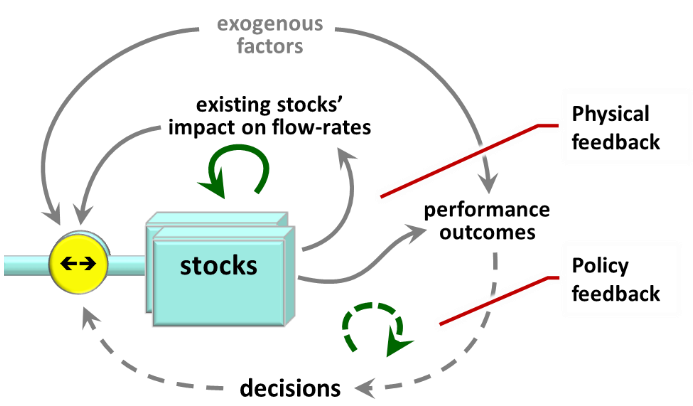
This diagram was captured, with estimated data-charts, in a half-day, and immediately identified that too much sales effort was going into seeking new prospects, rather than converting those prospects into paying customers. That same session also identified that technician effort was being taken up on appraising prospects' assets, rather than installing the company's devices in those assets to generate revenue. However, that problem would ease as customers in the segment were won, so only a small increase in technicians would be needed. A working model, including division of the client-base into 3 major segments, was completed in the following three weeks by a novice modeller, and used to plan priorities for which market segments to target, and which customers within each segment, as well as the size and deployment of the sales force and technical staff.

The system's core "physics", policy feedback and model boundaries

By this point in the agile process, the core interdependencies between the stocks in the system are clear and supported by numerical evidence – what might be termed the "physics" of the system plus the *physical* feedback it implies. So the final step is to complete the *policy* feedback, identifying what information is used, and how, to inform what decisions. Those decisions in the IT-support firm were the initial plan to add more clients and the hiring rate. The decision to hire the sales person and go for growth was driven by the firm's previous success, both in terms of good quality client support and profitability. Profitability is not shown in Figures 9 and 10, but is easily calculated from the fees paid by clients minus the costs of the staff and other items. The resulting model, including this policy feedback is at http://sdl.re/m912.

The generic structure that captures the principles followed in this agile process is shown in figure 12, where the initial iterations explain performance-over-time in terms of changing stock-values, driven by their own flow-rates, which are in turn driven by exogenous factors, decisions, and the existing stock quantities. This last dependency gives rise to the physical feedback in the system, to which is added the policy feedback driving the decisions (figure 12, dashed lines).

Figure 12: Functional and policy feedback, decisions and exogenous factors



Advantages of the agile process

Two general and important insights have arisen from repeated use of this process. First, improvements to performance are entirely focused on the flow-rates in the system, and for a simple reason. If performance of the system is entirely dependent on the quantity of its stocks, and exogenous factors remain constant, then zero flow rates mean no change in performance. Changes in performance *must*, therefore, arise from non-zero flow-rates.

The second insight follows directly from the first. Decisions must change performance by altering the flow-rates. Indeed, a great many decisions directly concern what those flow rates should be (client acquisition and hiring, in the IT-support case). Others, such as price changes, marketing spend or training, alter the flow rates less directly. This indicates a further problem with the more common process of model development, since CLDs rarely highlight either flow-rates or decision-factors.

The agile procedure in Figure 2 deals with most of the challenges found in the commonly proposed process in Figure 1. Since participants constantly see the need for numerical evidence – if only plausible estimates – they identify entities with which they are familiar, and whose definition and values are well understood. Since they also see the quantitative change in those entities being caused by quantitative changes in the factors on which they are believed to depend, they can immediately check that the causality makes sense and is in fact correct. Differences of opinion or understanding do not easily survive this process, so the problem owners and other stakeholders not only arrive at a shared mental model, but one that is demonstrably valid and usable.

Actionable insights arise very early and continue throughout this process. The IT-support case above is too simple to demonstrate the importance of this finding, because mapping the framework and building the model were completed in a matter of hours (although another day was needed to check the real data matching management estimates). For larger, more complex cases, however, model development may take longer and require periods of data-discovery. In such cases, it is useful to discover, for example, that a sales decline is due to a fall in customer numbers rather than lower average purchase rates, or that staff numbers are not rising because of increasing turnover, rather than a failure to hire.

The agile process substantially shortens the time needed to build out the model, not simply because the modelling is being done in parallel with the quantitative diagramming (possibly in the same room, at the same time), but also because data needs are identified early and can be fulfilled while other work continues. Furthermore, data that challenges stakeholders' beliefs regarding causality can be reflected immediately, so structural problems do not live on through the process.

The process reduces the need for large-group discussion and therefore both the demands on the time of busy personnel and the elapsed time over which the process occurs. The initial diagnosis may be carried out with the team who own the problem, rather than engaging all who may have some knowledge of any part of the wider system. Some of those others may become involved as the build-out of the model runs into issues on which they have knowledge, and it is useful to start such consultation by briefing those individuals on the model's progress to date. At each stage, however, the simple, logical and data-supported steps of the agile process take less time than the open-ended consultation needed to develop CLDs.

Since problem owners readily understand what is required and why, they can trust the modeller to work with those who have specific knowledge on particular points and need only reconvene to see the results of adding and quantifying those points. Engaging with the full range of stakeholders may be useful at the end of the modelling work, so that the entire audience shares the understanding to which they contributed. It is perfectly possible, even at this late stage, to correct any errors and add additional significant structure that may have been missed.

To be clear – qualitative causal loop or influence diagrams do not feature *at any stage* in this process. The procedure exposes feedback mechanisms that are evidenced by the data, and so long as the behaviour of concern is well-explained by the analysis, there is no reason to expect that additional feedback mechanisms are significant. Nevertheless, it may be legitimate to enquire as to whether additional feedback *could* arise, or *could* be designed into the structure to improve its management. Such enquiry, though, should follow the same logic as the foregoing analysis – asking which stocks in the system need to change their time-path of growth or decline, and at what rate; what changes to their flow rates need to happen for that to arise; and what interdependencies not previously identified might bring about that change? This enquiry may be valuable, for example, when seeking to improve policies or to redesign the system, such as with the addition or removal of stocks.

The process also identifies the relevant boundaries of the problem-space. These boundaries are discovered when no unanswered questions remain, rather than by trying to guess where those boundaries lie at the outset. Model boundaries may therefore be more limited than would otherwise have been the case, but could include additional scope arising from important mechanisms that the more common process missed through lack of attention to data.

Standard cases: standard systems

It was noted earlier that successful system dynamics professionals may not always, if ever, follow the common procedure described in the field's key sources and summarised in Figure 1. Two pieces of evidence support this supposition. First, leading academics and successful consulting groups in system dynamics tend to specialise in certain fields of application – Ford (2009) in water and power resources, Moxnes (2004) in renewable resources, Paich and Peck (2009) in pharmaceuticals, Cooper, Lyneis, Els, Ford and others (Lyneis and Ford, 2007) in project management, and so on. (*This is not to imply that these individuals could not or have not carried out other types of work; merely that they have a cumulative body of work in those domains*). Such professionals surely do not start each new enquiry with a blank sheet, and seek to build qualitative, shared mental models from each fresh set of stakeholders whilst bringing no known structures or phenomena from previous experiences.

The second piece of evidence is the recurrence of structures consisting of multiple connected stocks and flows carrying the same material from state to state. Many cases are simple chains – juvenile and adult fish stocks, staff-development chains, or susceptible-infected-recovered populations, for example – but others are more complex (see for example Carter and Moizer, 2011; Paich, Peck and Valant, 2011). The qualitative feedback structure of such chains is quite distinctive, consisting of multiple balancing loops, but are rarely evident in published CLDs of shared mental models. It is highly implausible that such structures could emerge spontaneously from the qualitative consultation process used to create shared mental models. Such structures must, therefore, have been brought to the project by the SD professionals themselves, rather than reflecting the mental models of stakeholders.

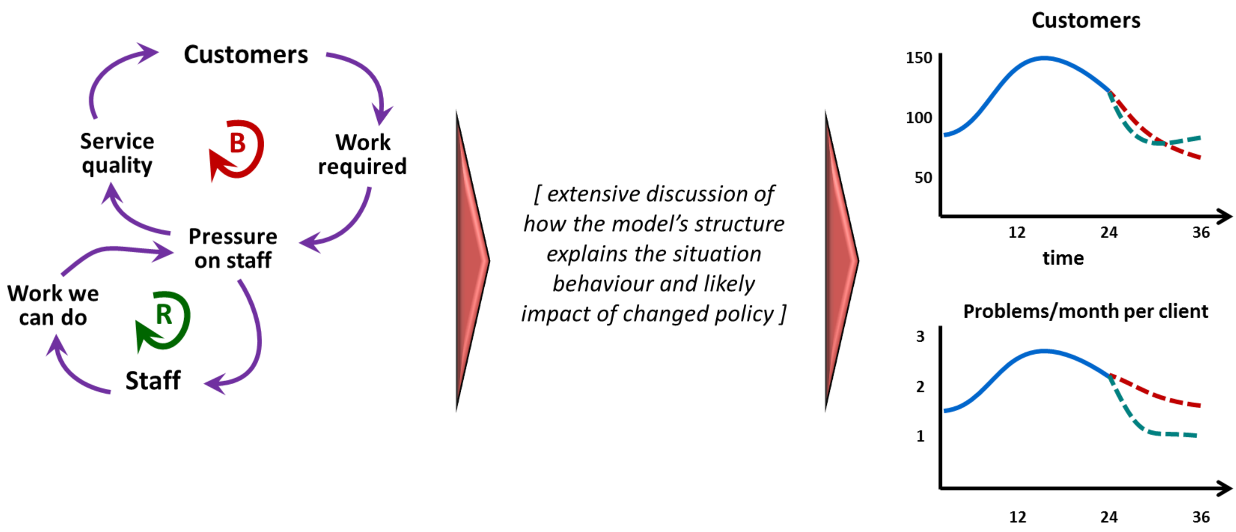
Practitioners in various domains, then, do not appear to start over with a new CLD-development effort for each new model project, but rather transfer knowledge and structures from case to case. This is possible because the *modus operandi* of the physical system in any given situation is very similar to other cases in the same domain, although the policy elements of the system may differ. Whatever the method by which the very first model in a domain was developed, subsequent projects have identified and developed standard structures that are found to be widespread. This principle should be extensible to many more domains than have been documented. Not only does every fishing region have fish, vessels and fishermen, but every law firm has clients, lawyers and cases; every health-care system has patients, medics, treatments and hospitals; every city has criminals, victims of crime and police. Many of the causal dependencies are also shared between similar cases – vessels catch fish and medics treat patients – so although the numerical values and strengths of relationships will differ markedly between different examples, the underlying physics will be shared.

The logical, self-validating agile process allows inexperienced professionals to achieve impactful work early in their SD career. For more complex cases, they can also start work from the standard templates developed by experienced colleagues. The stature and experience required to facilitate qualitative, multi-stakeholder debate is no longer critical. Those proven cases can also make clear to consumers of SD work exactly what they are going to get from it, and confidence that the result will work and be helpful, because it has been so for other users.

Presenting system dynamics models

The principles supporting the agile process also have implications for the presentation of SD. The final step in the common modelling process requires presenting model results and policy recommendations back to the stakeholders involved (and in the case of published work, to other audiences also). Presentations at the international conference, most articles on SD projects and most books, articles and slide decks are dominated by data-free diagrams – not only pure CLDs, but also stock and flow diagrams. Quantified results are most often presented in total isolation from the causal structure from which they emerge, requiring a leap of faith by the audience that those results do indeed emerge from the diagrams the experts have presented, based on the qualitative beliefs of the stakeholders. Figure 13 illustrates how this might be accomplished with a CLD representation of the IT-support case.

Figure 13: Separate presentation of model structure and performance implications



This presentational practice, too, is problematic. First, if the key aim of such presentations is to show how the system structure causes the resulting behaviour, the separation by many pages or slides between the display of structure and behaviour severely obstructs that purpose. Achieving the aim of building understanding of that connection surely requires at minimum that structure and behaviour be displayed together.

Secondly, it has already been noted that people cannot intuitively understand the behaviour of even the simplest feedback structure, so even if the qualitative structure and its behaviour are displayed together, it will be beyond the mental capacity of most readers to comprehend how the two relate to each other.

Even if the audience can follow or figure out how the structure might result in the broad behaviour pattern shown, the structure's qualitative presentation gives no sense of the scale of the causal mechanisms involved. In Figure 13, for example, did the explanation captured by the causal loop diagram at left find that the company had 5 more clients than it could serve, or 50 more, and was it short of 2 staff or 20? Should the hiring rate have been doubled or tripled, and over what period?

Since the findings from the process, and the resulting action plans must be quantified, every diagram communicating that explanation should also be quantified. Furthermore, because our concern is with how performance changes *over time*, that diagrammatic explanation must also communicate the dynamics of factors along the causal pathways. Combined with the accepted reality that no-one can intuitively estimate behaviour from qualitative causal structures in any case, this implies that SD diagrams should ***never*** be shown without time-chart data. Of course, diagrams that include time-charts on every single item quickly become cumbersome, but charts should at least be shown on key items in the structure, such as the stocks and their flows.

Thirdly, since the analysis was incremental and the problem to which it was applied perhaps extensive and complex, the explanation of the findings must also be incremental, with each of the steps in Figure 2 being explained in turn and each section of the resulting model presented in turn, as shown in Figures 3 to 8. Although CLDs are often built up in a similar incremental manner, many communications of SD projects present only a whole-system diagram, supported by extensive verbal explanations – quite beyond the comprehension of those not familiar with system dynamics.

Lastly, mental models rarely highlight those factors on which stakeholders have discretion to decide or to act. Not a single article has been found from the last 10 years of System Dynamics Review, for example, that highlights decision-items in the CLD. Subsequent discussion of such CLDs may refer to decisions and policies in general terms, but that is not the same as making clear on the model itself the location of decisions that may allow problem owners to influence system behaviour.

Conclusions

Concern across the system dynamics field regarding the slow adoption and limited impact of the method, compared with the considerable relevance and value it could potentially offer, has now reached a level that calls into question the very means by which the method is practised and communicated. Such questioning suggests that the process implicitly recommended by key sources, and followed in the training for young professionals contains such substantial flaws that the process should be fundamentally rebuilt, starting from the most basic principles of the method.

Following those principles in practical cases has led to a process that bears striking similarity to the "agile" method now used in many software development projects. Solutions are built hand-in-hand with users, and working models and actionable insights delivered continually. Experience over many years with the agile process described in this paper suggests it is fast, effective in delivering actionable insights, and reliable in generating similar structures for similar cases. Nevertheless, the method should be seen as a radical *adjustment* to group model-building processes (Vennix, 1996), rather than as a replacement, collapsing the time and effort required by possibly as much as an order of magnitude.

Most modelling projects carried out with the method to date have concerned relatively simple cases, so more work is needed to test the method's suitability to larger scale and more complex challenges. Fortunately, many such large, complex cases have already been tackled by experienced professionals in the field, resulting in well-known, repeatable structures that are applicable (with modification) to a large number of cases in the domains where they are relevant.

Together, the agile process for modeling new problems and the use of proven structures for well-known situations may deliver valuable system dynamics models, faster and more reliably, and thus contribute to greater adoption and impact of the method in the real world. The two procedures could have the incidental but significant benefit of accelerating young professionals' mastery of the science and practice of system dynamics, adding further to its adoption.

Abrahamsson P, Salo O, Ronkainen J and Warsta J. 2002. *Agile development Methods: Review and Analysis*. VTT Publications **478**.

Barlas Y. 1989. Multiple tests for validation of system dynamics type of simulation-models. *European Journal of Operational Research* **42**(1): 59–87.

Barlas Y. 2014. Unpublished comment, noted in a presentation to the Asia-Pacific Conference of the System Dynamics Society in Tokyo, February 2014.

Carter D, Moizer JD. 2011. Simulating the impact of policy on patrol policing: introducing the emergency service incident model. *System Dynamics Review* **27**(4): 331-357.

Coyle RG, Exelby DR. 2000. The validation of commercial system dynamics models. *System Dynamics Review* **16**(1): 27–41.

Cronin M, Gonzalez C, Sterman JD. 2009. Why don't well-educated adults understand accumulation? A challenge to researchers, educators and citizens. *Organizational Behavior and Human Decision Processes* **108**: 116–130.

Cronin M, Gonzalez C. 2007. Understanding the building blocks of system dynamics. *System Dynamics Review* **23**(1): 1–17.

Fisher D. 2010. *Modeling Dynamic Systems*. ISEE Systems: Lebanon NH.

Ford A. 2009. *Modeling the Environment: An Introduction to System Dynamics Models of Environmental Systems*. Island Press: Washington DC.

Forrester J, 2013, Economic Theory for the New Millennium, *System Dynamics Review* **29**(1): 29-41.

Forrester JW, Senge PM. 1980. Tests for building confidence in system dynamics models. In *System Dynamics: TIMS Studies in the Management Sciences*, Legasto AA, Forrester JW, Lyneis JM (eds). North-Holland: Amsterdam: 209–228.

Forrester JW. 2007. System dynamics—the next fifty years. System Dynamics Review 23(2–3): 359–370.

Fowler M. 2001. *Is Design Dead?* in Succi G and Marchesi M (eds). *Extreme Programming Explained.* Addison-Wesley: Boston.

Franco LA, Rouwette EAJA. 2011. Decision development in facilitated modelling workshops. *European Journal of Operational Research* **212**: 164–178.

Graham AK. 1980. Parameter estimation in system dynamics modeling. In *Elements of the System Dynamics Method*, Randers J (ed). Productivity Press: Cambridge, MA: 143–161.

Haselton MG, Nettle D, Andrews PW. 2005. The evolution of cognitive bias. In DM Buss (Ed.), *The Handbook of Evolutionary Psychology*. Wiley: Hoboken NJ.

Homer J, 2013, *The aimless plateau, revisited: why the field of system dynamics needs to establish a more coherent identity,* System Dynamics Review **29**(2): 124-127.

Huz S, Andersen DF, Richardson GP, Boothroyd R. 1997. A framework for evaluating systems thinking interventions: an experimental approach to mental health system change. *System Dynamics Review* **13**(2): 149–169.

Kahneman D, Slovic P, Tversky, A. 1982. *Judgment under uncertainty: Heuristics and biases*. Cambridge University Press: Cambridge MA.

Lane DC. 1994. *Modeling as Learning: A Consultancy Methodology for Enhancing Learning in Management Teams.* In *Modeling for Learning Organisations*. Morecroft JDW, Sterman JD (eds). Productivity Press: Portland OR.

Lane DC. 2008. The emergence and use of diagramming in system dynamics: a critical account. *Systems Research and Behavioral Science* **25**(1): 3–23.

Lyneis JM, Ford DN. 2007. System dynamics applied to project management: a survey, assessment and directions for future research. *System Dynamics Review* **23**(2): 157-189.

Maani KE, Cavana RY. 2000. *Systems Thinking and Modelling*. Pearson Education New Zealand: Aukland.

McCartt AT, Rohrbaugh J. 1995. Managerial openness to change and the introduction of GDSS. *Organizational Science* **6**: 569–584.

Milstein B, Homer J, Hirsch G. 2010. Analyzing National Health Reform Strategies With a Dynamic Simulation Model. *American Journal of Public Health* **100**(5): 811–819.

Morecroft JDW. 1982. A critical review of diagramming tools for conceptualizing system dynamics models. *Dynamica* **8**(1): 20-29.

Morecroft JDW. 2007. *Strategic Modelling and Business Dynamics*. Wiley: Chichester UK.

Moxnes E. 2000. Not only the tragedy of the commons: misperceptions of feedback and policies for sustainable development. *System Dynamics Review* **16**(4): 325-348.

Moxnes E. 2004. Misperceptions of basic dynamics: the case or renewable resource management. *System Dynamics Review* **20**(2): 139-162.

Paich M, Peck C, Valant J. 2009. *Pharmaceutical Product Branding Strategies: Simulating Patient Flow and Portfolio Dynamics.* Informa Healthcare: New York NY.

Paich M, Peck C, Valant J. 2011. Pharmaceutical market dynamics and strategic planning: a system dynamics perspective. *System Dynamics Review* **27**(1): 47-63.

Paich M, Sterman JD. 1993. Boom, bust, and failures to learn in experimental markets. *Management Science* **39**(12): 1439-1458.

Peterson DW, Eberlein RL. 1994. Reality checks: a bridge between systems thinking and system dynamics. *System Dynamics Review* **10**(2–3): 159–174.

Professor Jay W Forrester, 2013, *Fireside Chat*, Conference of the International System Dynamics Society, July 24th.

Pruyt E. 2013. *Small System Dynamics Models for Big Issues*. TU Delft Library: Delft, Netherlands.

Richardson GP. 1986. Problems with causal-loop diagrams. *System Dynamics Review* **2**(2): 158–170.

Schwaninger M, Groesser SN. 2009. System dynamics modeling: validation for quality assurance. In *Encyclopedia of Complexity and System Science*. Meyers RA (ed). Springer: Berlin: 9000–9014.

Scott RJ, Cavana RY and Cameron D, 2014, Evaluating immediate and long-term impacts of qualitative group model building workshops on participants' mental models, *System Dynamics Review* (forthcoming).

Sterman J. 2000. *People Express Management Flight Simulator (2000 edition)*. Strategy Dynamics Ltd: Princes Risborough: UK.

Sterman JD. 1989. Modeling managerial behavior: misperceptions of feedback in a dynamic decision making experiment. *Management Science* **35**(3): 321-339.

Sterman JD. 1994. Learning in and about complex systems. *System Dynamics Review* **10**(2): 291-330.

Sterman JD. 2000. *Business Dynamics*. McGraw-Hill: Boston MA.

Sterman JD. 2010. Does formal system dynamics training improve people's understanding of accumulation? *System Dynamics Review* **26**(4): 316–334.

The Balanced Scorecard Institute. 2013. *Building & Implementing a Balanced Scorecard: Nine Steps to Success*. https://balancedscorecard.org/Resources/TheNineStepstoSuccess/tabid/58/Default.aspx; Retrieved 16-3-2014.

The International Council on Systems Engineering (INCOSE). 2011. *Systems Engineering Handbook v3.2.2*. INCOSE: Los Angeles CA.

The Supply Chain Council. 2013. *The SCOR Framework.* https://supply-chain.org/scor: Retrieved 16-3-2014.

Vennix JAM, Scheper W, Willems R. 1993. Group model building: what does the client think of it? In *Proceedings of the 1993 International System Dynamics Conference*. System Dynamics Society: Chestnut Hill, MA.

Vennix JAM. 1996. *Group Model Building.* Wiley: Chichester UK.

Wolstenholme EF. 1994. *A Systematic Approach to Model Creation*. In *Modeling for Learning Organisations*. Morecroft JDW, Sterman JD (eds). Productivity Press: Portland OR.

Yücel G, Barlas Y. 2011. Automated parameter specification in dynamic feedback models based on behavior pattern features. *System Dynamics Review* **27**(2): 195–215.