Abstract for:

LESSONS ON MODELING FROM THE SYSTEM DYNAMICS NATIONAL MODEL

Alan K. Graham
Research Associate
System Dynamics National Project
Alfred P. Sloan School of Management
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Since 1972, Jay Forrester and colleagues at MIT have been evolving the System Dynamics National Model (SDNM). The purpose of this model is to guide policy makers in dealing with today's major economic problems. The ambitious scope of the project motivates careful examination of modeling practices and how they contribute to success of the project. The above paper recounts incidents in the development of the SDNM and discusses the related modeling issues. The paper suggests these guidelines:

1. Statements of purpose for a successful modeling project must become more and more sharply-focused as the project proceeds. If the purpose remains diffuse, the modeling will not produce results, and may exhibit the "kitchen sink" syndrome, where extensive structure is added in high hope but for no good reason.

2. Divide the model into meaningful sectors, preferably standardized, to facilitate the process of model development and testing.

3. Complex models are treacherous and can conceal their flaws. The proper attitude toward complex models is suspicion and distrust. The modeler should try to force the model to reveal its flaws as soon as possible, and in the simplest possible configuration.

4. Test individual components first, and gradually combine them into larger and larger configurations, taking care to understand the behavioral implications of each added piece of structure. Avoid working with only the full model for extended periods of time.

5. Problems in pieces of a model are hardly ever solved by adding additional pieces. Problems should be analyzed in the simplest possible configuration in which they can occur.

6. Be pessimistic about the ability to understand how parameters and structure give rise to behavior of a complex model. This implies maintaining a set of full and partial configurations where at any one time, one knows about the behavior of each and the relations among them. The alternative is getting lost in parameter and formulation space.

7. In testing a model, divide time evenly between in-depth examination of single simulations and experimenting with different parameters and configurations. It is productive to keep an eye on pedestrian details such a table overruns, equilibrium conditions, long-term behavior, and step responses in several inputs.

8. Avoid parameter tuning. Approach behavior problems from the point of view that the model structure is flawed. Without that viewpoint, it is very difficult to detect flawed structure, and very easy to engage in lengthy and fruitless parameter tuning.

9. Avoid quick formulation fixes. Thoroughly review the entire ensemble of conditions to which the proposed formulation is supposed to respond. The alternative is often handling three quarters of the conditions ten times.

10. Decide among formulations by testing rather than administrative fiat. The tests can be either simulations or preferably thought experiments.

11. Every piece of the model should show plausible response, even if inputs from other pieces are not plausible. This gives a model that is both realistically robust and easier to debug.

12. Avoid months-long individual modeling efforts. Stay in constant contact and trade off tasks on a week-by-week or even day-by-day basis.

13. Use the telephone for day-to-day coordination.

14. Use computer messages for complex technical discussion, especially about structure. The messages build an archive, and so should be both concise and complete with references to earlier messages and specific computer runs. Computer messages allow both arguments and rebuttals to be well-thought out, uninterrupted, and permanently recorded.

15. Use face-to-face meetings for longer-term planning, brainstorming, examining computer runs, and reaching consensus (after the appropriate groundwork has been laid).

16. If preliminary model results make good sense independently of the model, releasing them both gives a better climate for funding and provides a sharper focus for further model work.

17. It is just as easy to be methodical about unimportant issues as important ones. The guidelines above favor a methodical approach; following them will contribute to the success of a project only if they are used in the context of actually producing useful results.

LESSONS ON MODELING FROM THE SYSTEM DYNAMICS NATIONAL MODEL

Alan K. Graham
Research Associate
System Dynamics National Project
Alfred P. Sloan School of Management
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

To be presented at the
1981 System Dynamics Research Conference
The Institute on Man and Science
Rensselaerville, New York
October 14-17, 1981

LESSONS ON MODELING FROM THE SYSTEM DYNAMICS NATIONAL MODEL

Alan K. Graham

Introduction

The System Dynamics National Model (SDNM) Project began in 1972 when Professor Jay Forrester was approached by a representative of the Rockefeller Brothers Fund to apply System Dynamics (SD) to economic issues. Previously, Forrester had produced two provocative studies, Urban Dynamics and World Dynamics (out of which grew the Limits to Growth report for the Club of Rome) [1-3]. Forrester accepted the challenge of studying the U.S. economy, funded for the first three years by the Rockefeller Brothers Fund. After the startup funding wound down, the present method of funding evolved: sponsorship by a coalition of corporations, foundations, and individuals, with occasional participation by government agencies. The sponsors meet every six months to hear current results. The project has produced interim results that are promising and insightful enough to generate continued funding.

Professional dynamicists are often uneasy about the SDNM. The early years of the project seemed to lack the sharply-defined model purpose that characterizes many other SD studies. The model often seems too complex for even skilled dynamicists to understand. And the project has gone on for a remarkably long time.

To a considerable extent, the duration of the project can be explained by comparing the small number of dynamicists working on it to the size of the task. Yet in retrospect, time has been lost tackling problems that could have been solved more quickly with different tactics. This paper addresses those tactical issues.

To provide structure for the discussion, the paper is organized by steps or activities in the model development process:

--model purpose
--designing model architecture
--testing
--revising
--communicating and chronicling
--releasing results

Experienced modelers will find no unfamiliar issues here; most of the issues are faced by every builder of large models. But much of the knowledge and experience for dealing with these issues are usually transmitted only through the oral tradition in System Dynamics. This paper records and organizes some of those discussions as they occurred within the context of the System Dynamics National Model Project. The intent is primarily to assist modelers in running a large modeling project, and secondarily, to satisfy some curiosity about the SDNM.

Model Purpose

A model's purpose is the ultimate basis for deciding what should and should not be represented in a model. But in many ways, a model's purpose evolves with the model itself. At the beginning of a project, the model purpose will be to deal with some set of fragmentary and diffuse initial issues: undesirable present conditions, undesirable past behavior, anticipated difficulties, or questions about what to do with a given set of policy levers. Only as the project continues will the modeler generate information that more sharply focuses the model's purpose: a reference mode (a formal statement of behavior to be changed), a dynamic hypothesis (an hypothesis about the structures that create the behavior and the symptoms), or a list of the policy instruments to be studied. Only when the modeling is well-advanced will there be a limited and integrated set of symptoms, reference modes, dynamic hypotheses, and policy levers.

Thus, the startup of a modeling project can take many forms. Many industrial projects started from a clear recognition of oscillatory behavior. Many others start with questions on what to do with a given policy lever. But *Urban Dynamics* began from a set of symptoms. And *World Dynamics* began from the concern that in the future, natural resources might not permit the world economy to reach the U.S. level.

The System Dynamics National Model (SDNM) began with concerns drawn more or less from the headlines: unemployment, inflation, economic fluctuations, low capital investment, impending natural resource shortages, overpopulation, the depression, uncertainty over monetary and fiscal policy, and inequities in regional income distribution, to name a few. Only as the project continued could the model's purpose sharpen. Different concerns turn out to have different relations to the model's purpose and behavior. For economic fluctuations there is a reference mode, a dynamic hypothesis, and even a separate model [4]. Inflation is seen as the consequence of price changes due to several behavior modes [5]. Some concerns such as overpopulation are not yet modeled. Some concerns such as regional income distribution are not planned to be addressed at all. Still other concerns have become more restricted, with concern about natural resource shortages becoming a modeling project on transition to alternative energy sources. Finally, some apparently unrelated concerns came to be seen as symptoms of a behavior mode not considered in the beginning: the economic long wave.

The idea of considering economies moving in 50-year cycles came out of the model development process. When the model first showed long waves, it was assumed that some flaw in the model produced them. But examination revealed plausible mechanisms producing the 50-year cycle. Apparently, it was the conception of how the real economy behaved that would have to change. Only afterwards did we begin to locate a literature and evidence for long waves in economic growth, interrupted by major depressions. The long wave hypothesis brought together several concerns (the depression, stagflation, and low capital investment) as well as bringing in several new concerns, not addressed at the beginning of the project (slowing productivity growth, low technological innovation, changing political attitudes) [6].

In brief, then, a sharply-focused model purpose is something that emerges from the modeling process. In this, the SDNM is much like any other SD model. But because of the scope of the project, it has taken years for some concerns to emerge into sharp focus.

How do inadequacies in model purpose reveal themselves? If the model purpose is beyond the resources of a modeler, modeling efforts fail to get nearer to the goal. If the model attempts to address concerns that are not related, the normal course of modeling will reveal the true connections among symptoms, behavior, and structure. Finally, if the model's purpose is vague and is allowed to remain so, there will be chronic indecisiveness on what should be represented in the model, or a chronic "kitchen sink" approach of including structure for no good reasons. The SDNM project has not shown any of the classic symptoms of inadequate model purpose: there is no lack of results being produced; some dynamic hypotheses have been wrong and then revised; and discussion on model formulation has not focused on <u>what</u> to represent, but rather on <u>how</u> to represent.

Designing Model Architecture

Only novices would begin structuring a model by starting to write equations directly. Smaller models may be started by identifying important loops and levels. Larger models are usually started by specifying an architecture that divides the proposed model into sub-models, customarily called sectors.

Dividing a model into sectors serves two purposes: First, the division provides a framework within which individual cause-and-effect relations may be located. Second, division into sectors provides a natural progression for testing: The sectors can be tested first individually, and then in increasingly larger configurations.

The architecture of the SDNM has been detailed elsewhere [7], so it will be discussed here only as it relates to more general modeling practice. The SDNM architecture evolved from earlier work on industrial systems; the core of the model is a generic representation of a corporation or an industrial sector (with allowance for aggregation). This generic production sector can be replicated an arbitrary number of times to represent an arbitrary number of sectors of the economy, such as a capital-producing sector, a durable goods sector, an agriculture sector and so on. Each replication can have its own parameters to reflect the differences among sectors regarding pricing, capital investment, hiring, and so on. Other sectors are then natural additions to this core: The household sector makes purchases and decides on workforce participation. The labor sector represents wage-setting and movement of people between sectors of the economy. The financial sector represents interest rate setting and loan activity. Other sectors are mapped out as well.

The SDNM views the economy as made up of sectors, many of which have structures similar to that of a single corporation. This view is vital to a workable testing strategy: each sector is smaller than the total model and has meaningful internal dynamics. The present version of the SDNM contains 5 sectors: capital, goods, household, labor, and financial. With standardized sets of parameter changes, any of these sectors can be inactivated, and its outputs to the other sectors held constant. Moreover, within each sector there are a variety of switches that make constant or inactivate money flows, prices, wages, interest rates, capital ordering, or hiring. The switches allow great flexibility in testing a variety of configurations. For example, one configuration might be a goods-producing sector receiving constant orders from the household, with prices, borrowing, and the effects of money flows inactive, and with labor, and physical capital available at constant price. A modeler could then change the configuration for example, to allow the price of goods to vary, and learn the effects of price on the internal dynamics of the sector. Testing a sequence of such configurations allows the modeler to "creep up" gradually on the full system, learn exactly where the problems first appear.

Standardization is another dimension of the SDNM architecture. Standardization is a strategy for avoiding duplication of effort; the generic production sector is a good example. When the present SDNM architecture was first conceived, up to 15 separate production sectors were visualized. Without standardization, one could imagine spending five years writing all 15 sectors separately, "inventing the wheel" 15 times. With a standardized production sector, however, the number of sectors remains flexible. By taking replications to represent highly aggregated sectors of the economy, it was possible to generate insights into economic behavior with only a small number of sectors very early in the project.

A bias toward standardization is a bias toward discovering generality. This meshes well with a long-standing System Dynamics tradition of attempting to deal with problems at the simplest and most general level first, and filling in details only as needed. A good example is the household sector of the SDNM. As a functioning sector of the national economy, the household sector performs many of the same functions as the producing sectors. The household acquires goods and services, handles money balances, and borrows. To exploit this commonality, the household sector is represented as another replication of the generic production sector, with some special equations to reflect the household not having an output product it sells to other sectors.

Standardization has been extended to within the individual production sectors as well. For example, there are generic ordering equations, which allow each production sector in principle to order an indefinite number of factors of production, such as labor, capital, energy or raw materials. Similarly, each production sector can incur both short-term debt and long-term debt, but there is only one debt equation, subscripted in the dimensions of debt category (short or long) and sector. Standardization allows the SDNM to be very concise. The DYNAMO III simulation language uses subscripts and arrays, so that (for example) any time anyone purchases anything from any part of the economy, that transaction is represented by a single vector equation in the SDNM [8]. Once the modeler learns something about one purchase, something is learned about all purchases. But it is quite possible to overstandardize. With enough conceptual contortion and sophistry, much more of the household sector could have been standardized. Similarly, at one time, we had implemented the acquisition of debt simply as the acquisition of another factor of production like labor or physical capital. However, in both cases, the standardization was working in the direction of conceptual complexity rather than simplicity, so they were abandoned.

Model Testing

Attitude toward model. How should one go about testing a model as complex as the National Model, with its many sectors and multiple behavior modes? The answer that seems to emerge from looking backward at the National Model Project seems to be that one tests the National Model much like any other System Dynamics model, but with extraordinary care to maintain the proper attitude toward the testing. Complex systems are treacherous, not only for participants in real systems, but also for modelers: flaws in a model may manifest themselves under some circumstances, but remain hidden in many other circumstances. Murphy's law holds for the National Model: anything that can go wrong will go wrong, at the worst possible moment. Thus, the proper attitude for testing is suspicion and distrust of the model.

With a suspicious attitude, one assumes that flaws exist and tries to force the model to manifest them as soon as possible. Indeed, many flaws have been detected first by plotting out fifty or a hundred variables and studying them aggressively to see which curves and relationships between curves might signal potential trouble. In practice, the most effective procedure for detecting flaws seems to involve splitting one's time roughly in half, divided equally between such in-depth examination and making a variety of runs exploring the model's response to different parameters and configurations.

What to test. To understand the behavior of a complex model, there must be a thorough understanding of the behavioral implications of the pieces. One comes to understand the system by gradually incorporating pieces, working with fuller and fuller configurations. Exploring fuller models has the same functions and pitfalls as an army sending people ahead to scout the terrain: some advance information is useful, but if everyone is out scouting, the main body itself has no substance.

As an example of the pitfalls of focusing testing on fuller models, consider what was discovered in testing the household as a separate sector. With the money and borrowing influences active, the purchase of goods showed an 8-year oscillation. The household would order goods, go into debt, and only later cut back on orders due to shortage of money and credit. The behavior revealed several implausible gains and delays in the household sector's money system. Had this behavior been scouted out in a fuller model, it probably would have been interpreted as evidence of too much of the employment-inventory interaction responsible for business cycles. The behavior could have been mitigated by parameter changes in the corporate sectors, but the result would have been business cycles arising from spurious causes. This incident suggests the necessity of thoroughly testing each sector separately.

If a piece of model misbehaves and the cause is not obvious, it is often a temptation to argue that the malbehavior arises because of absence of structure not yet incorporated. Philosophically, that approach is quite close to the classic "kitchen sink" approach: "surely if we have enough structure, the right links will be there, and the right behavior will emerge." Historically, both approaches have been disastrous. Adding more structure usually adds to malbehavior. At best, the flaws within the smaller model are temporarily hidden. As a rule, it has been more productive to confront misbehavior where it occurs, or in even simpler configurations.

When one has already tested the individual pieces of a large model, and is in the process of combining those into larger configurations, it is not particularly exciting to go back and retest the individual components after formulation and parameter changes have been made. Yet if testing stays on a larger model and continues over a period of several months with formulation and parameter changes, the model slowly drifts into an unknown region in parameter and formulation "space" where the origins of the larger model's behavior are no longer known with confidence. It has happened more than once that, after several months of working with a large system, we have gone back to test the individual sectors and found behavior nothing like that at the start. The large model's behavior arose from mechanisms quite different than what we thought.

For example, we evolved a large model that sometimes gave a vigorous long wave, but it was extraordinarily sensitive to parameters and initial conditions. It turned out that the behavior of the individual sectors was remarkably unlike their (previously realistic) behavior prior to the extensive changes done within the large model. The behavior of the large model had been tuned through parameter and formulation changes in the apparent direction of realistic behavior. That behavior turned out to be due to a set of mechanisms that after the fact seemed largely spurious. Yet because most variables were correlated into a strong long-wave mode, the spurious mechanisms were very difficult to detect in the fuller model. The only symptom of difficulty was an extraordinary sensitivity to parameter values and initial conditions. It has been argued theoretically that parameter tuning can be misleading [9], and practice corroborates the argument.

Rather than focusing the testing on a single full model for long periods of time, it has been more fruitful to change the configuration being tested fairly frequently, or have different people work on different configurations. That way, for example, one is never very far from knowing what the relationship of the behavior of a single capital sector is, relative to the behavior of a two or three sector model. As one continues to shift among several configurations, one begins to see consistent patterns of how adding given pieces of structure affects the behavior. For example, activating wage change usually affects the long wave by increasing its amplitude and period. One can thereby work backward from the full system (about which empirical information is available) to arrive at standards for what the behavior of the individual sectors should be, in order to result in realistic behavior by the full system. Also, when one shifts between configurations fairly rapidly, one avoids the problem of having to analyze model flaws in large models where it is difficult to distinguish between symptoms and causes. One goes back to the simplest possible configuration where the problem can be manifested, and tackles it in that simpler setting.

<u>How to test</u>. The philosophy [10] and methods [11] of testing SD models have been discussed elsewhere; the remarks here add specificity to those more general discussions.

Very much in contrast with other disciplines, the primary use of statistical information in the SDNM is to invalidate some behavior modes. For example, the SDNM has been plagued by a very strong oscillation whose period ranged between 11 and 20 years. Employment in the capital-producing sector would quadruple within 5 to 10 years, and then reverse that growth within another such interval. The fluctuation originated from an interaction between capital pricing policy and capital investment policy. From our exploration of the macroeconomic statistics, we are convinced that such behavior is unrealistic. This conviction, based on statistical evidence, has been responsible for a great deal of testing, analysis, and revision. Thus, the principle role of macroeconomic data has been of a negative sort, invalidating some behavior modes rather than validating others.

The majority of "tests" that first revealed model flaws have been mundane: when there are table overruns, first asking whether the table is behaving properly even under those extreme conditions, and second asking why the input is at such an extreme. As an example of the payoff in aggressively pursuing such pedestrian matters, a great improvement in the model's robustness was produced by changing from the TABHL function to the extrapolating TABXT function in several places, so that under extreme conditions, changes in the input could always have some effect, even though small: previously, once the input moved beyond the range of the TABHL function, the input made absolutely no difference, so that whatever feedback loops it was involved in were effectively cut.

As another mundane test, when variables in equilibrium do not reach their neutral values (1.0 for multipliers, for example), ask how that can be. Asking that question proved quite productive in the case of the formulation that ordered physical capital and controlled the amount already on order. For many inputs, the ordering formulation could not reach a neutral equilibrium. For example, under most circumstances, the ordering formulation would not bring the amount of capital on order to equal what was defined as the desired amount. So equilibrium values would depend on the amount of feedback that would result from such discrepancies, which in turn would depend on time constants, slopes of table functions, and so on. It was as if the model were representing an organization where some conflicts were never resolved, so the outcome would depend on which party pushed hardest. There were other difficulties with the formulation as well, but the activity that resulted in a satisfactory formulation was trying to create a formulation that could reach a sensible equilibrium. This was finally achieved, and the model became substantially more robust as well.

There are a variety of other tests equally mundane and equally productive. Implausible phase relationships can reveal improper gains. Very long simulations can reveal long drifts and inappropriately small gains, or the lack of absolute standards to determine an equilibrium. Implausibly large parameter sensitivity can reveal lacking or incorrect feedback structure. Even simple step input tests can be quite revealing if one uses several runs to test steps into several inputs: not only steps in demand for output, but also in capital or labor price, interest rates, or loan availability.

Revising

The DYNAMO compiler makes it much easier to change parameters than structure; if misbehavior can be eliminated by changing parameters, it is tempting to do so. The model has often entered into pathological states severe enough to abort the simulation, and these crashes could be eliminated by parameter changes. But stepping around a bear trap doesn't remove the bear trap. Likewise, changing parameters usually doesn't remove the flaws that created the pathological behavior; usually, the flaws are only hidden, waiting to reveal themselves in some other situation.

In a related phenomenon, it has happened often that one can adjust parameters to give one behavior mode very nicely, but at the expense of eliminating some other behavior mode in a different configuration. If one approaches such difficulties with the view that they can be solved by manipulating the parameters, one is trapped into a lengthy process of changing parameters to achieve some behavior modes, but simultaneously losing others. Most likely, there are structural flaws which no amount of parameter tuning will cure.

Only by approaching misbehavior with the view that the structure may be at fault can one detect structural errors. Curiously enough, starting from an examination of structure has more often than not yielded conceptual insights about parameter values. But that is the frosting; the cake is that one becomes much more efficient at finding opportunities to improve the model through structural changes. And typically, structural changes reduce the need for parameter tuning. In fact, it has become a very useful discipline to stick to one set of parameters for several months at a time. While not precluding sensitivity testing, this discipline does tend to steer researchers away from parameter tuning and toward examination of the structure to achieve the desired behavior.

In revising model structure, the "quick fix" is dangerous. The SDNM is a large, complex model; its formulation must be designed to respond correctly to a variety of extreme inputs and combinations of inputs. Typically, even a formulation that exhibits a flaw for one set of inputs already has been successfully designed for correct response for several other sets of inputs. A "quick fix" usually ends up correcting the new flaw, but at the expense of losing the ability to respond correctly to other inputs.

The alternative to the "quick fix" is immediate and aggressive review before any proposed changes are implemented. This is an opportunity for correcting simple mechanical oversights, as well as the more fundamental task of recollecting the other sets of inputs to which the proposed formulation should respond. This recollection constitutes a complete specification for what the proposed formulation should be able to do. The process is time consuming and frustrating, but considering all of the inputs thoroughly once has proven to be more efficient than considering three quarters of them ten times: Without a full inventorying of the complete set of specifications, a vicious cycle sets in, where a new formulation works for a while, but then has difficulties in some new configuration, which motivates going back to the original formulation. There are several places in the SDNM where the formulation has gone back and forth several times before a more comprehensive and durable formulation came out of a comprehensive review.

Every modeling project is under pressure to produce results, and the National Model, which is supported only by its success in the marketplace of ideas, is no exception. There are many instances when a modeler is confronted with a choice between two formulations that are apparently equally plausible. Due to pressures to produce results, there is a temptation to simply declare the issue a modeler's choice, pick one, and move on. Choice by fiat does have some legitimacy when the alternatives are stylistically different but mathematically identical. However, choice by fiat has often led to troubles further down the road when applied to two formulations that are fundamentally different. A formulation that works well in business-cycle dynamics can fail later when the long wave is being tested.

It sometimes happens that when alternative formulations are being compared, each has some advantages and some disadvantages. At that point it may be premature to make a choice at all; it has been very productive to press further in the research for alternatives. Sometimes, more creative work can yield a formulation that not only lacks the disadvantages of the original alternatives, but is simpler as well. Such formulations are worth the effort, even if they can't be found in all cases.

A modeler can evaluate a formulation either by actually simulating it, or by thought experiment. Of the two, thought experiments are generally more discriminating. Modelers can imagine a much greater variety of extreme inputs and combinations of inputs than a model can generate at any one time. And if there is a flaw in the formulation, it is at least ten times more efficient to detect it before it is ever put into a model.

Thought experiments work best for small fragments of structure, where the input-output relationship is analyzable. By contrast, thought experiments do not do well in predicting how a given structural change affects overall model behavior. There is no way around the need to simulate. But any one simlation will generate only part of the many behavior modes and extreme conditions of which the model is capable. Many structural changes in the SDNM have mitigated this limitation by having a facility for switching (parametrically) between the old and new formulations. These switches allow the modeler to re-experiment with the alternatives, possibly several months later, in a different configuration, with a different behavior mode, and with different combinations of inputs. Of course eventually the unused formulations are edited out of the model.

In creating equations for a model that must produce several distinct behavior modes, it is a challenge just to keep all the modes in mind. It is sometimes all too easy to neglect robustness -- the ability to behave realistically even under extreme conditions or parameter changes. More specifically, it is desirable to have each piece of the model behave plausibly, even if the inputs are implausible. In other words, each piece of the model should protect itself against mistakes in the other parts of the model. Not only is robustness usually a realistic feature, but also it can accelerate model development.

Consider the case where a mistake in one part of the model can propagate to all the others: when one comes to debug such a model, one must carefully examine the entire model. In the example earlier, the flat TABHL function propagated erroneous behavior throughout the entire model, by cutting off a realistic feedback in the capital ordering process. By contrast, if each piece of the model can protect itself, the only implausible response would be where the implausible response originated. At one point recently the SDNM model was reviewed and reformulated explicitly for robustness. At least by subjective impression, afterwards malbehavior has been much easier to ascribe to a particular piece of the model.

Communication and Chronicling

In a small academic organization, everyone wears several hats: supervisor, laborer, administrator, planner, salesman, writer, outside consultant, and teacher. For the most part, schedules are chaotic, and no one can devote extended uninterrupted periods of time to working on the model. It is a special organizational challenge when the fundamental activity being organized is as solitary and individual as modeling. During the beginning of the project, researchers simply worked on separate sectors of the model. Although this procedure produced highly polished individual sectors, the isolation of the individuals working was mirrored by the isolation of the sectors; even today, several sectors developed in Ph.D. theses remain unintegrated with the existing SDNM. Extensive personal forays into modeling specific areas do not seem to produce the well-integrated structure and careful focus on major issues that are desiderata of the SDNM.

A much more effective strategy seems to be working as a team, staying in close touch and trading off tasks on a week-to-week, or sometimes even day-to-day basis. This procedure keeps a vigorous pace in advancing the fully integrated model. People's motivation and focus on the importance of the work remains high. Moreover, there is synergy when two people working in distinct areas both begin to suspect troubles in the same formulations: two vague instances of uneasiness can produce one fairly certain identification of a problem. Also, staying in frequent contact (usually by phone) allows a rapid shifting of attention onto the questions that seem most important to the overall project. Finally, the task of communicating insights forces both continual critiquing and concise expression. For these reasons, having the research team stay in close contact has proven to be very desirable.

There are varieties of close contacts. It has taken several years for the present research group to learn when to use various media. There are several available to researchers at MIT: face-to-face meetings, telephone conversations, computer messages (sent via out time-sharing consoles), and typewritten papers. The latter are hardly ever used, since computer messages are usually much faster.

There has been a tendency to rely too much on verbal discussion, especially in discussing alternative formulations. Verbal discussion is easily sidetracked, and impermanent. It is difficult to inventory the full set of inputs to which a given formulation is supposed to respond. It is difficult to extemporaneously create the appropriate thought experiments that can discriminate among formulations.

Face-to-face meetings are useful for situations where group discussion is required: planning who should undertake what, deciding a formulation question (after the appropriate groundwork has been laid), or "brainstorming" where the group creates several ideas on which individuals can later follow up. Meetings are also useful if computer runs need to be examined and discussed by several researchers.

Telephone conversations have many of the same advantages and disadvantages as meetings, so phoning has many of the same uses: planning, deciding, and brainstorming. But the telephone is more convenient than meetings, so the bulk of the day-to-day coordination is done by phone.

The most useful medium for exchanging complex technical information, especially about structure, has been computer messages. They are permanent, concise, and complete. One can easily reference earlier discussions of the same issues. One can give rebuttals at length without detracting from the original presentation. Information is available to many people quickly, without waiting until everyone can be at one place at one time. Turnaround delays in dictating, transcribing, proofing, revising, copying, and distributing are eliminated. Computer messages are sendable and receivable either at home or at the office. They communicate and record exactly what conditions were when a modeler obtained a given result. (This is useful for reconciling different results from different people.) Finally, if the research group expects periodic messages, each researcher is forced to be both explicit and concise about what problems are being solved and what is being learned from simulations. As a rule, a full day of simulations seems to result in about one 8 1/2" X 11" single-spaced page of summary. Usually, one sends messages only every 2-3 full days of simulations, with the number of pages multiplied accordingly. Thus the amount of writing involved is minor compared to simulating and analyzing.

Releasing Results

A variety of people have expressed a concern that releasing results before a project is finished is unscientific. This concern is puzzling, since it is rare both in the scientific community and in the system dynamics community to wait until definitive results are achieved. Biologists publish what they have learned from an experiment, even though it is one in a series. Computer scientists deliberately publish preliminary designs, in order to be reviewed before the designs are finalized. System dynamicists often release preliminary findings well before the project is over, and often the project and the client both benefit: A project as ambitious as the SDNM could not have generated continuing enthusiasm and support without release (in some form) of preliminary results. In addition preliminary release has improved the quality of results: Explaining model behavior to nontechnical audiences has forced an evolution of the explanations and evidence for behavior in a way that may not have happened otherwise.

Of course it is possible to abuse the process of releasing preliminary results. One temptation is to release findings solely on the basis that the model generates them. But a model is always imperfect and experiments can be flawed. One cure for that temptation, which is rigidly followed in the SDNM project, is to require that preliminary findings make good sense on their own merits, independent of the model.

Conclusions

Rather than focusing exclusively upon the System Dynamics National Model, this is perhaps the place to emphasize the lessons that have been learned as they apply to all large modeling projects.

1. Statements of purpose for a successful modeling project must become more and more sharply-focused as the project proceeds. If the purpose remains diffuse, the modeling will not produce results, and may exhibit the "kitchen sink" syndrome, where extensive structure is added in high hope but for no good reason.

2. Divide the model into meaningful sectors, preferably standardized, to facilitate the process of model development and testing.

3. Complex models are treacherous and can conceal their flaws. The proper attitude toward complex models is suspicion and distrust. The modeler should try to force the model to reveal its flaws as soon as possible, and in the simplest possible configuration.

4. Test individual components first, and gradually combine them into larger and larger configurations, taking care to understand the behavioral implications of each added piece of structure. Avoid working with only the full model for extended periods of time.

5. Problems in pieces of a model are hardly ever solved by adding additional pieces. Problems should be analyzed in the simplest possible configuration in which they can occur.

6. Be pessimistic about the ability to understand how parameters and structure give rise to behavior of a complex model. This implies maintaining a set of full and partial configurations where at any one time, one knows about the behavior of each and the relations among them. The alternative is getting lost in parameter and formulation space.

7. In testing a model, divide time evenly between in-depth examination of single simulations and experimenting with different parameters and configurations. It is productive to keep an eye on pedestrian details such a table overruns, equilibrium conditions, long-term behavior, and step responses in several inputs.

8. Avoid parameter tuning. Approach behavior problems from the point of view that the model structure is flawed. Without that viewpoint, it is very difficult to detect flawed structure, and very easy to engage in lengthy and fruitless parameter tuning.

9. Avoid quick formulation fixes. Thoroughly review the entire ensemble of conditions to which the proposed formulation is supposed to respond. The alternative is often handling three quarters of the conditions ten times.

10. Decide among formulations by testing rather than administrative fiat. The tests can be either simulations or preferably thought experiments.

11. Every piece of the model should show plausible response, even if inputs from other pieces are not plausible. This gives a model that is both realistically robust and easier to debug.

12. Avoid months-long individual modeling efforts. Stay in constant contact and trade off tasks on a week-by-week or even day-by-day basis.

13. Use the telephone for day-to-day coordination.

14. Use computer messages for complex technical discussion, especially about structure. The messages build an archive, and so should be both concise and complete with references to earlier messages and specific computer runs. Computer messages allow both arguments and rebuttals to be well-thought out, uninterrupted, and permanently recorded.

15. Use face-to-face meetings for longer-term planning, brainstorming, examining computer runs, and reaching consensus (after the appropriate groundwork has been laid).

16. If preliminary model results make good sense independently of the model, releasing them both gives a better climate for funding and provides a sharper focus for further model work.

The precepts above have a bias toward the methodical that can be dangerous if carried too far. For a research project to be successful, it delivers results on time, but the results must resolve fundamental problems. These two requirements will often conflict, and the conflict will be mirrored among the researchers as well. The quick-acting and result-oriented will often be at odds with the methodical. Roberts [12] has indicated that certain personalities and roles are necessary for a successful R&D team, so it is not farfetched to suggest that the two types of researchers are necessary in an SD project. It should be emphasized here that experience has shown that it is just as possible to be methodical in unimportant areas as in important ones. So the precepts above will be useful only when used within the context of actually achieving useful results.

References

[1] Forrester, J.W., Urban Dynamics, MIT Press, Cambridge, Massachusetts (1968).

[2] Forrester, J.W., World Dynamics, MIT Press, Cambridge, Massachusetts (1971).

[3] Meadows, D.L., et al., Limits to Growth, Universe Books for Potomac Associates, Washington, D.C. (1972).

[4] Mass, N.J., Economic Cycles, MIT Press, Cambridge, Massachusetts (1976).

[5] Forrester, J.W., Controlling Inflation, M.I.T. System Dynamics Working Paper D-3282-1 (July, 1981).

[6] Graham, A.K. and P.M. Senge, A Long-Wave Hypothesis of Innovaton, Technological Forecasting and Social Change 17: 283-311 (1980).

[7] Forrester, J.W., N.J. Mass, and C.J. Ryan, The System Dynamics National Model: Understanding Socio-Economic Behavior and Policy Alternatives, Technological Forecasting and Social Change, 9: 51-68 (1976).

[8] Pugh, A.L. III, DYNAMO III User's Manual, MIT Press, Cambridge, Massachusetts (1981).

[9] Graham, A.K., Parameter Estimation in System Dynamics Modeling, TIMS Studies in the Management Sciences, 14: 125-142 (1980).

[10] Senge, P.M., and J.A. Bell, Methods for Enhancing Refutability in System Dynamics Modeling, TIMS Studies in the Management Sciences 14: 61-74 (1980).

[11] Forrester, J.W., and P.M. Senge, Tests for Building Confidence in System Dynamics Models, TIMS Studies in the Management Sciences 14: 209-228 (1980).

[12] Roberts, E.B., Generating Effective Corporate Innovation, Technology Review, 27-33 (November, 1977).