STELLA: Software For Bringing System Dynamics To The Other 98%

Barry Richmond
Dartmouth College

ABSTRACT

A revolutionary new piece of software called **STELLA** is introduced. **STELLA** -- which stands for Structural Thinking, Experiential Learning Laboratory with Animation -- is a system dynamics "expert system," embodying expertise in the areas of computational and structural logic, conceptualization, equation formulation and model analysis. **STELLA** is designed to facilitate efforts to bring the system dynamics framework to a much broader audience. The software enables even those lacking computer experience or a quantitative orientation to conceptualize, construct and analyze high-quality system dynamics models, while accelerating the development of an intuition for dynamics. Several **STELLA**-based products targeted at audiences in both formal education and business are described.

INTRODUCTION

For several years now a question has haunted system dynamicists. It has been asked in several ways, but as yet, no widely agreed upon answer has been given. The question is: *If this stuff really is so great, then why hasn't the field "taken off"?* I would like to suggest that the primary reason that the field has not taken off is that we have not been clear enough about what it is that is "so great" about system dynamics, and that in the absence of such clarity, we have not devoted enough attention to what is really necessary to ignite such a takeoff.

In this paper, I first will advance a hypothesis which seeks to explain why system dynamics has fallen far short of its potential penetration into the marketplace of ideas. Next, I will describe the key features of a new piece of software, called **STELLA**, that is designed to spur the widespread penetration of the system dynamics framework. The use and implications of **STELLA** both within, and outside of, a formal education setting are treated next. Finally, I will close with some conclusions and suggestions for an agenda for system dynamicists.

WHEREFORTH ART OUR TAKEOFF ?!

Our field has not taken off because we have not figured out how to effectively and efficiently transfer the *process for doing* system dynamics. It takes a very good student at least two full-length courses, a supervised project, and usually a thesis involving some significant model-building effort, just to achieve *first-level* proficiency in system dynamics. To become "good" currently takes another couple of years of apprenticed practice. How many people have this much time to devote to developing a set of skills that can help them do whatever they really want to do even better? And, how many proficient system dynamicists are there with the time to service the needed apprenticeships? I would offer the same answer to both questions:

"not many!". Hence, as long as we continue to rely on system dynamics courses and apprenticeships as the primary vehicles for transferring our process, the field will *never* take off!

Let's step back for a minute. System dynamics has been practiced for more than twenty-five years. Why, in all this time, have we not developed an efficient and effective means for transferring the process? One argument, sometimes advanced, is that: we have indeed developed such a means, it's just that system dynamics is such a subtle art that it takes many years to fully master. This argument may have some merit. But, even so, certainly it is not an excuse for abandoning efforts to reduce the time that it takes to come up to full speed. I feel that an important part of the reason why we have failed to significantly improve the means for transferring the system dynamics process is that we have been confused as to exactly what the *real* value-added of our discipline is. In particular, too many of us have been seduced into thinking that the real value-added lies in the *products* of our process--such products include specific models, as well as the insights emanating from these models. It is easy to see how such a perception could come about. Our best known books and papers are about "products", not process. And, there certainly is a bigger payoff, both in academe and business, for providing insights and "answers", than there is for illuminating a method for arriving at good questions.

Payoffs and publications notwithstanding, I feel that the *real* strength of system dynamics lies in its ability to enhance, enrich and extend the learning experience. Using system dynamics, people can learn faster, achieve a deeper level of understanding, and retain the resulting knowledge for a longer period of time. The fact that particular system dynamicists have used the process to arrive at particular insights in particular domains of study is interesting and important, yet somewhat beside the point. I am not suggesting that system dynamicists stop thinking about matters in other disciplines, or that we should arrest all production activities. I am suggesting that products from our process are *not* the most important things that we have to offer the world, and further that such products--so long as they are perceived as coming from "outsiders"-- always will meet with substantial resistance by those within the given field--no matter how good the documentation is! If we wish to spread our discipline, the way to do it is through widescale transfer of the process, not through dissemination of our products. It is from this viewpoint that the motivation for **STELLA** springs.

STELLA: A SYSTEM DYNAMICS "EXPERT SYSTEM"

As noted above, there simply are not enough qualified professionals available to teach system dynamics courses, even if we did have an effective and efficient means for transferring the process. One way around such a shortfall in labor is to substitute capital. Enter **STELLA**, a system dynamics "expert system". "Expert systems" is a branch of the newly-emerging field of "Artificial Intelligence." An "expert system" is the embodiment of some type of expertise in a computer program. The program then is used by less-skilled practitioners both to produce more "expert-like" results, and to further the development of their own expertise. Expert systems in medicine, sales, manufacturing, engineering, oil & gas exploration and mining now are in use, and many new ones (in a wide variety of fields) currently are under development. **STELLA** is the first attempt at a system dynamics expert system.

STELLA is an expert system in the sense that it is software that embodies knowledge of structural and computational "rules", as well as model-creation heuristics. This knowledge can help even those with little or no computer expertise, and lacking a quantitative orientation, to conceptualize and build high-quality system dynamics models. A set of simulation techniques and analysis aids, also built into the software, are designed to help accelerate the development of an intuitive feel for dynamics, and to guide users in performing a high-quality analysis of model behavior. Let us briefly examine each of these embodied features in turn.

In using **STELLA**, a person builds a "structural diagram" -- a more accurate name than "flow diagram"-- by selecting structural elements from a "tool kit" positioned along the left-hand margin of the screen of Apple's Macintosh personal computer (see Figure 1). Once selected, each element is positioned on the screen, and then "hooked up" as desired to reflect the structural relationships under investigation. Embodied in **STELLA**'s computer code is a set of structural and computational rules which relieve the user of much of the time-consuming, technical tedium of building and debugging a model using a traditional computer language (like DYNAMO).

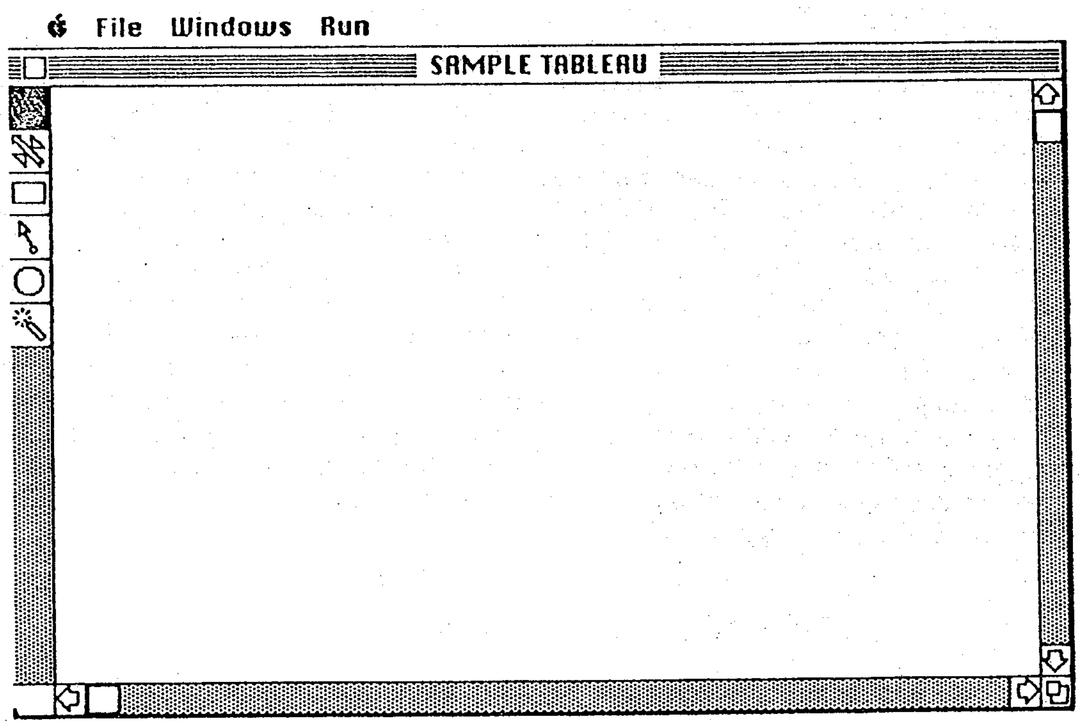


Figure 1. A Sample Tableau with "Tool Kit."

For example, any time that a "level" is positioned on the screen, the difference equation structure, representing the algebra associated with the level, is *automatically* generated. If the user wants to inspect these equations, it is possible to do so by obtaining an equation listing. But there is *never* a need to write them! Once a "flow and flow-regulator assembly" are attached to the level, the algebraic direction of flow is automatically established (the arrowhead points the direction of *positive* flow). "Clouds" are automatically generated where needed.

Circles are used to store equation logic. In **STELLA**, the circle is viewed as an integral part of the valve assembly. This is consistent with the notion that auxiliary variables are algebraically substitutable into rate equations. Before the circle is attached to the valve assembly, the assembly is "dumb"--just a passive, mechanical device. Attaching the circle adds "intelligence". Hence, in **STELLA**, there is no conceptual distinction recognized between a circle that is attached to a valve assembly and one that is not. Both are receptacles for logic. One uses the logic to translate information inputs into outputs that regulate a rate of flow. The other uses the logic to generate outputs that are used for other purposes. Within this conceptual framework, "constants" are seen as circles which have no inputs--thereby eliminating the need for an additional structural diagramming symbol.

Equation logic is entered by "opening" the circle icon with a double-click of the mouse button. In entering the logic: (1) the software "understands" the timing-of-calculation sequence, so that no timescripts need be used, (2) any variables that are *allowable* in the equation automatically are present within the "opened" circle, and can be inserted into the equation by "clicking" on them--no need to retype the variable name, and hence no chance for typos!, (3) set-aside "special functions" (like steps, ramps, etc.) also are automatically available, and also are insertable via "clicking"-- eliminating the possibility of misplacing needed arguments (like step height, step time, etc.), and (4) "graphical functions" (called "Table Functions" in DYNAMO) can be drawn using the mouse--the numerical coordinates associated with the particular curve being sketched are computed and displayed as the sketching is taking place (see Figure 2).

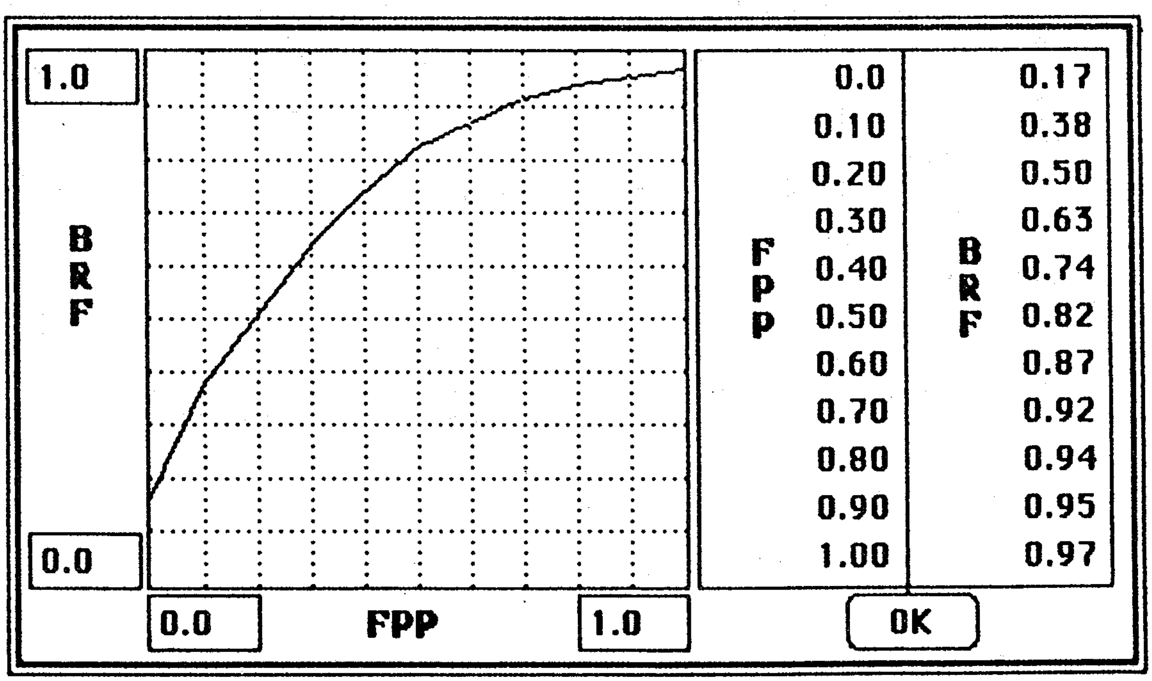


Figure 2.
A Sample Graphical Function: Birth Rate Fraction as a Function of Food Per Person.

If a user shows "n" inputs to a particular circle on the structural diagram, the software is smart enough to put those same "n" inputs on the list of allowable variables for the equation defining the circle (see Figure 3). If the user tries to include some other quantity (not shown as an input on the structural diagram), **STELLA** knows not to allow it. Conversely, if an equation has been defined, and then later the user adds another input(s) to the structural diagram, the software will light up a "?" in the associated circle to indicate that the defining equation no longer is in one-to-one correspondance with the structural diagram. **STELLA** displays additional "intelligence" by flagging simultaneities. If, in seeking to link circles (using information arrows), simultaneity is encountered, the software will not allow such a linkage to be made (i.e., simultaneity is identified while it is being created, rather than making a person work to "track it down" after a disappointing, abortive attempt to compile). Finally, the software also knows not to allow information arrows to "stick" to levels, or flow assemblies to "stick" to circles.

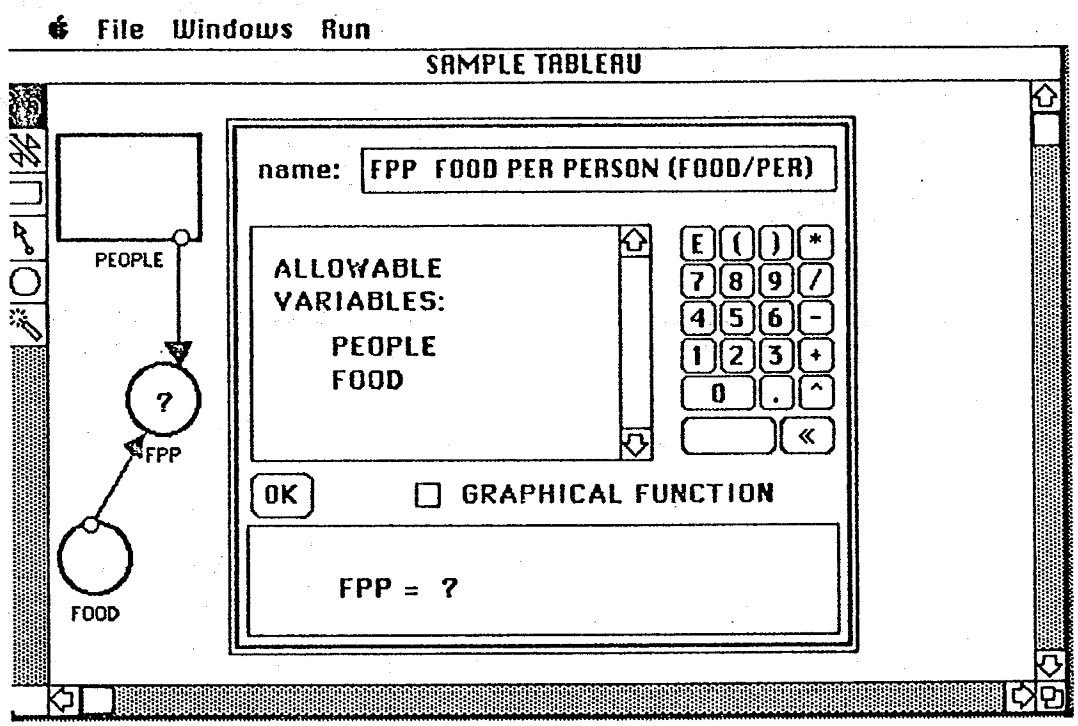


Figure 3. A Window Showing the "Inside" of the Circle "FPP"

By making the software "smart enough" to embody several such structural and computation rules, even people just starting out in system dynamics will find it easy to avoid making "dumb" mistakes. And, freed of this demoralizing burden, the resulting savings of time and effort then can be devoted to learning the more important aspects of system dynamics.

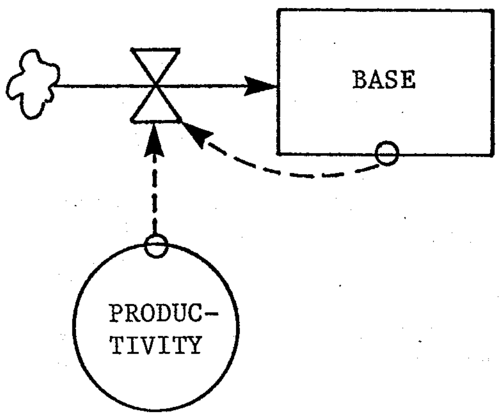
In addition to "smarts" with respect to structural and computational rules, **STELLA** also embodies "intelligence" for aiding in the conceptualization and equation formulation processes, as well as some simulation techniques that are designed to facilitate the development of a person's intuition for dynamics and analytical capabilities.

STELLA is designed with features that encourage people to conceptualize their models as a set of stocks or accumulations. For example, a "zoom" feature enables a person to view the large sheet of "electronic paper", upon which the structural diagram is created, as a matrix of miniaturized pages. The user then can "spray" the matrix, placing one reduced-in-size stock onto each miniaturized page. Next, by telescoping in to full size on each page--one at a time--the flow processes associated with each stock can be added. In doing so, **STELLA's** accompanying documentation encourages people to make use of one of the five generic flow processes depicted in Figure 4. These processes have been distilled from years of experience in building system dynamics models in a wide variety of contexts, and can be used to represent most commonly encountered flow processes. The generic structures can be stored in an "electronic scrapbook", dumped onto the screen, and then appropriately parameterized when needed.

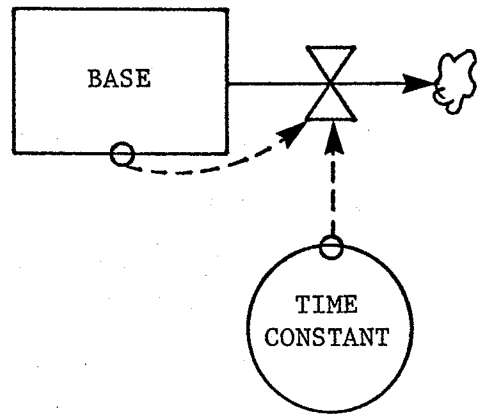
The availability of these generic "atoms of structure" is another **STELLA** feature that makes it much easier for a novice to arrive at a well-formulated model. By employing these set-aside "atoms", the new-comer's effort will not be expended in reinventing some already well-known formulation for a flow regulation process (like a birth rate, a retirement rate, or a production function). Instead, the person will be freed up to be creative about the quantity, type and shape of inputs to a birth rate fraction, an average residence time, or a productivity term. Once the flow processes associated with a given stock have been added, another **STELLA** feature enables the model-builder to easily "tab" to the next page in the matrix, and there to repeat the process: level, then associated rates of flow.

In this approach, the stock becomes the conceptual "unit of analysis". The model is built one stock at a time. And, once each stock has been appropriately outfitted with flow processes, zooming back to an intermediate distance enables the modeler to begin deciding how to suture together the page-distinct stock/flow assemblies. **STELLA** facilitates users in rearranging their tableaus by permitting these assemblies to be lassoed and then repositioned as desired. Freely bendable, stretchy information arrows can be used to link structural elements in the desired manner. A "ghost" of any element also can be created in cases where stretching an information arrow would be too cumbersome, or graphically unasthetic.

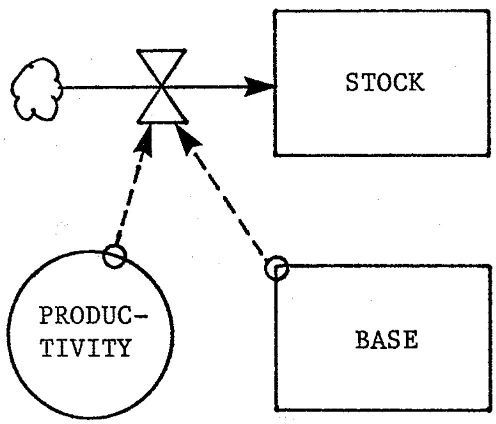
At any time during the process of model construction, the user may elect to view model behavior. There is no lengthy pause to await compile. And, it is not necessary to have specified DT, plot period, print period, or length in advance of requesting a simulation. **STELLA** has default values for each of these parameters built in (they are: 0.05, 1.0, 1.0, and ∞, respectively). Furthermore, if the modeler has left one or more rates undefined, **STELLA** simply will assume a value of zero for each and proceed (rather than generating a fatal error message). All of these features make it easy for even the computer-naive person to retain a feeling of being in control during a computer session. And, with **STELLA**, at the end of a simulation session, there is no need to shift modes to resume model construction. The distinction between "edit" and "rerun" is non-existent.



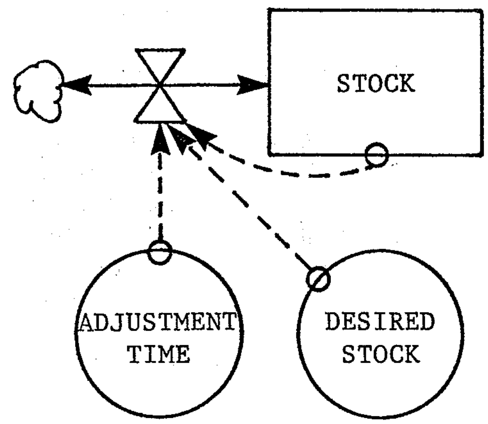
1. Self-Generating Production Process



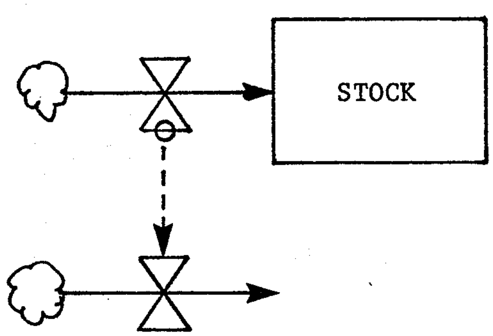
2. Self-Generating Flow-Thru Process



3. Non Self-Generating Production Process



4. Explicit Goal-Seeking Process



5. Co-Flow Process

Figure 4. Five Generic "Atoms of Structure"

In addition to the typical time series plots and tabular prints commonly available for displaying simulation output, **STELLA** includes *animation* and scatter plot capabilities. In the animation mode, the dynamic behavior being generated by the model is superimposed on the structural diagram. Stocks fill up and deplete, flow speeds are registered on dials, and the values taken on by graphical functions appear as readings on pressure guages. And, **STELLA** enables people to zoom in and out--i.e., from macro "reporting variables" down to the microstructure of an individual rate-controller--while animation is unfolding. In addition, it is possible to pause the animation to examine the particular configuration of stock magnitudes and graphical function pressures that exist at any point in time. All of these features enable people to develop an intuition for dynamics much more quickly than if simulation output were displayed only by static time series graphs or printed tables of numerical values. Experience with STELLA thus far has confirmed that most people can store and recall mental images of reservoirs building up and depleting much more readily than pictures of static curves traced out on multiple axes against time. The availability of scatter plots (x vs y) augments **STELLA's** animation capabilities (see Figure 5). Being able to watch how one variable is moving with respect to another over the course of a simulation further enhances the development of a person's intuition for understanding dynamics, and facilitates the analysis of model behavior.

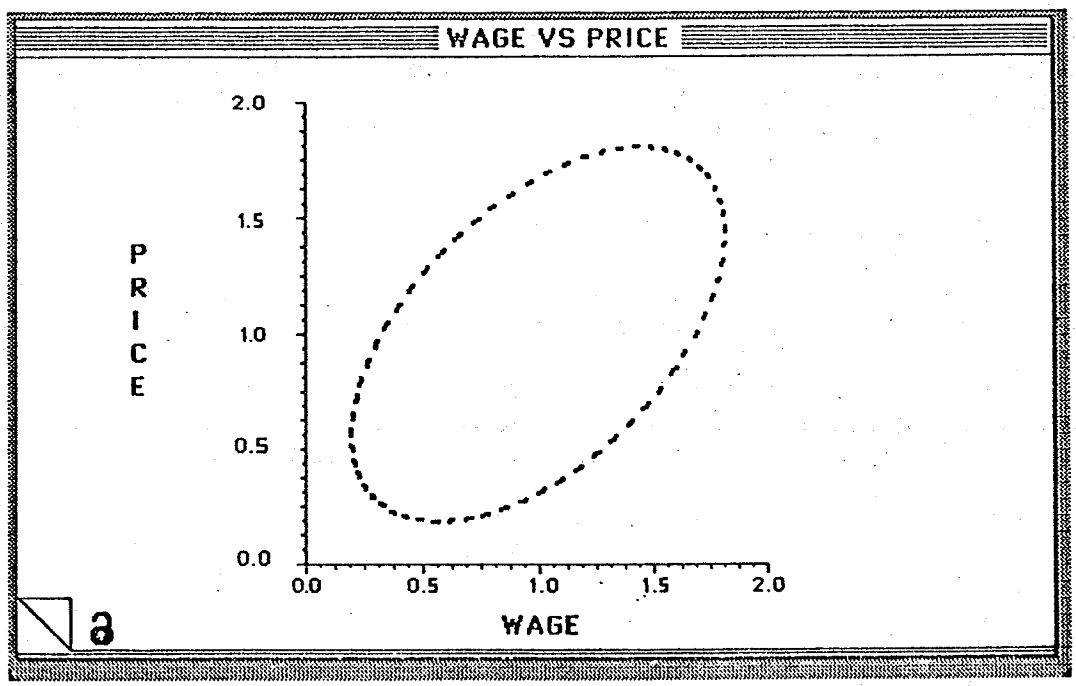


Figure 5. An Illustration of a STELLA Scatter Plot.

These, then, are the salient features of a system dynamics expert system. To date, in its prototype form, **STELLA** has been used with a variety of audiences varying in age, experience with computers, and substantive background. A few examples will help to illustrate the potential of the software. Using **STELLA**, students in an introductory system dynamics course (which I have taught for six years) were at least 50% more productive in learning the course material than their counterparts from previous years. Both the speed of learning *and* the depth of understanding were improved significantly. I have incorporated **STELLA** into an ongoing seminar series that I conduct for a Boston-area high-tech firm. Since doing so, I have witnessed several managers construct and analyze simple (3-4 level), but good quality, system dynamics models after only three days of exposure to the concepts and approach. In each case, model construction activity actually began after only *one* day's exposure! Prior to using the software, course participants were unable to engage in *any* formal modeling activity on their own. One final example is provided by an experience with a completely computer-naive group of senior managers from a large financial institution. In two extended sessions, the group used **STELLA** to help identify several key reorientations in their strategy for managing a new consumer-oriented investment product. Prior to this exposure, individuals in the group had had no previous exposure to computer simulation.

These examples are indicative of the potential of expert system software such as **STELLA** for greatly increasing the transferability of the system dynamics process. With care and attention in the design of the software, this increased rate of transfer can be achieved without compromising quality. But now the question arises as to how we can go about making the best use of this enhanced potential for transferring our process. Over the last year or so, a small group of people have been hard at work seeking answers to this question. The papers to be presented at this Conference by Peter Vescuso and Steve Peterson serve as fruitful testimony to the success of these efforts. In the next two sections I will lay out some general thoughts as to how STELLA might best serve the end of bringing system dynamics to a broad audience.

STELLA IN A FORMAL EDUCATION SETTING

If you accept the major premise of this paper--which is that the *real* value-added of our discipline lies in its use as a "learning utility"--then system dynamics should be one of the principal frameworks by which high school and college students learn what they learn. In order to "bring system dynamics to the other 98%" in a formal educational setting, it is essential for *us* to go to *them*. And, the most promising "thems", I would suggest, are in the social sciences and humanities.

System dynamics can be especially useful in the social sciences and humanities because learning there is gaited by the absence of a laboratory environment. The availability of laboratory facilities provides a major learning advantage to students in the physical sciences and engineering. In a laboratory setting, students literally can "grab hold" of the abstractions presented in the classroom. They can work "hands on" with the material, and can do so at their own pace and in accordance with their own learning style. And, perhaps most importantly from a learning standpoint, they can engage in *controlled experimentation*. Such experimentation often is not practical, and sometimes not even physically possible, in the social sciences and humaniities. Computer simulation offers the opportunity for creating such a laboratory environment for students in these fields. To date, the problem has been that many students in the social sciences and humanities do not have the technical skills and quantitative orientation needed to interact effectively with computers. **STELLA** obviates these needs. To make effective use of this software requires creativity and conceptual rigor, not great mathematical prowess or computer skills.

Thus far, two types of **STELLA**-based "learning laboratories" have been conceptualized, with first-examples of each currently under development.

The first is a series called "Learning Laboratories In: . . .", where the ". . ." is filled in with the name of particular substantive-area disciplines like microeconomics, physics, anthropology or psychology. The notion behind the series is to *add value* to the "industry standard" treatment in a given discipline, rather than to reframe the material in some fundamental way. The papers by Steve Peterson and Peter Vescuso should provide ample illustration of the specifics of the approach. I will briefly discuss the general outline.

Substantively, the table of contents of each book in the series will read much like those which appear in the well-established texts in a particular field. The treatment of each topic within the field will add value to the current standard treatment by providing a laboratory environment which will render the material dynamic, less abstract, and structurally/conceptually more rigorous. Each chapter will be organized as a sequence of lab experiments, each dealing with a key set of concepts. Each lab will build on preceding sessions, and each will be progressively less structured. The "experiential learning" approach reflected in the laboratories will consist of two kinds of activities: discovery (or production), and investment. In the discovery process, two techniques will be employed. The first will present students with a configuration of structural elements and ask them to hypothesize what behavior the configuration will generate. The second technique will present students with a behavior and then ask them to synthesize a structure capable of generating that behavior. The latter technique should prove especially useful for enabling students to discover "gremlins" (i.e., departures from the theoretical ideal, like friction, imperfect competition and other realistic, but usually less mathematically tractable, phenomena). In the investment process, the same two techniques will be employed. Students will be given a structural element, like a "graphical function", and be asked to parameterize it to reflect their experience or knowledge. Alternatively, they will be given an axis and asked to sketch a behavior that their knowledge/experience tells them is "correct".

Use of the experiential learning laboratory approach should greatly enhance, enrich and extend the learning experience. And, very importantly from a system dynamics viewpoint, the approach will put learning *under the control of the learner, rather than shifting the burden to the teacher!* In addition, because the "Learning Laboratories" series departs from, but significantly augments, the standard treatment in a given discipline, it will appeal to students *directly* rather than having to wait for faculty adoption and subsequent "trickle down". Once students who use the series begin to raise challenging questions, to demonstrate an increased depth of understanding, and to demand a more realistic treatment of the subject matter, faculty members will have great incentive to assimilate the "experiential learning laboratory" approach into their curricula.

The second **STELLA**-based learning laboratory series is called "Great Thinkers In:...", where the title again is completed with the names of various substantive-area disciplines. I currently am working on one for Economics, and will use it to illustrate the general approach upon which the series is based. In this series, too, the fundamental thrust is to proceed by *adding value* rather than beginning anew. In this instance, an excellent base of departure exists in the form of a book entitled The

Worldly Philosophers. In the book, Robert Heilbroner marches systematically through the theories of a set of great economic thinkers beginning with Adam Smith and concluding with John Maynard Keynes. Although the only quantitative data in the book are the page numbers, Heilbroner beautifully lays out the structure of each theory in prose that is very conducive to the construction of system dynamics models. By rendering the prose more structurally explicit, and then bringing the theory into the laboratory, students can substantially enrich and extend their understanding of these great economic syntheses. Through hands-on experimentation, students can get much closer to the material than is possible by the passive involvement provided by reading. Indeed, they can begin to *discover* the theories for themselves, rather than simply accepting them as given and commiting them to memory. Such experiential learning serves as a far more powerful substrate for the production of creative thought, and decays with a far greater time constant, than traditional "book learning". Another advantage resulting from bringing the great thinkers into the lab is that it then becomes possible to explore the linkages between them. For example, a student might discover that it is possible to get from Adam Smith's synthesis to the Reverend Malthus' theory by relaxing only one feedback loop, or that Mill and Marx are only a few state variables apart. All of these learning opportunities become possible once the computer can be made readily accessible as a learning laboratory.

To implement this Series, at least initially, likely will require participation by a system dynamicist. For example, at Dartmouth, I am pursuing conversations about co-teaching "Great Thinkers" courses with faculty in the Departments of History, Economics, Philosophy and Government. Where a well-recognized "Great Thinkers" book, like Heilbroner's, already exists, the need for a system dynamicist to be actively involved is reduced. The perceived threat posed by such courses is minimal, and the potential to add to the knowledge base is enormous.

Over time, as students make use of the "Learning Laboratories" and "Great Thinkers" series in a variety of courses, they will begin to develop that skill much talked about in system dynamics circles: the ability to see "transferabilities of structure". When birth rates, panic and market penetration are seen not as somehow fundamentally distinct, but rather as different manifestations of a generic "self-generating production process", systems thinking will have arrived. When students can see how the theories of some great economist, physicist and historian really are a lot more alike because of their underlying structure of assumptions than they are different because of superficial distinctions in language, we will begin to have the kind of people that are needed to address the major "systems" problems of our time--problems like acid rain, world famine and the arms race .

STELLA OUTSIDE OF FORMAL EDUCATION

STELLA can be used in many ways to bring system dynamics "to the other 98%" outside of a formal educational setting. The principal difference in the approach in these arenas that is implied by the software is a shift from viewing ourselves as purveyors of answers and insights, to seeing ourselves as facilitators and "transferers" of the process.

Traditional system dynamics consulting usually consists of the system dynamicist constructing, and then making available to the client, some reasonably large simulation model. The degree of involvement by the client with the finished-product model ranges from "virtually none" ("just give me the answers") to "quite substantial" (the client learns how to run--and analyze output from--the model, and may even go so far as to learn how to add or modify structure). This approach has several advantages, and in many cases meets a very real need in the marketplace. However, so long as clients remain "consumers" of *our* products rather than "producers" of their own, an important opportunity for client learning will go untapped.

Much of the learning that occurs when using the system dynamics approach occurs *during* the conceptualization and structure-development processes. The finished product can, of course, be used to generate significant additional learning. But it is clarifying the problem behaviors of interest, wrestling with what to make stocks and flows, ensuring that the dimensions are balanced, and discovering the feedback loops that were absent in the original mental model, that add most to one's stock of knowledge about a system. And, it is the clients who are most in need of adding to their knowledge base. What system dynamics provides is a way to make these knowledge bases more productive, to unlock the creative potential stored in them. What we have lacked is a way to get system dynamics to the client. **STELLA** can help in addressing this deficiency.

As I noted previously, a **STELLA** prototype has been used in a variety of professional contexts with great success. It *is* possible, as Jay Forrester has long envisioned, for CEO's and Board Members to build and exercise formal models. And I believe that in many cases they are the people who *should* be doing it! The kinds of models that such people will build are not likely to be of the large, operationally-oriented variety. Instead, they probably will be small and strategic in their orientation. I now have witnessed four or five such models come into being. The largest had six levels. The smallest had only one. But all precipitated a tremendous amount of good discussion and learning. And each profoundly altered the way a group of people looked at their organization. These results *never* could have been achieved had the simple models been delivered as "fait accomplis". They would have been ridiculed as "too simplistic".

There is a tremendous opportunity outside of formal education for changing the way people think and view the world. The opportunity is exciting, both intellectually and commercially. Software, like **STELLA**, can greatly aid in realizing this opportunity. The possibilities are limited principally by your imagination.

CONCLUSIONS AND PROPOSED AGENDAS

The real power of system dynamics lies in its ability to serve as a *learning utility*. To date, only a very small fraction of this power has been realized. **STELLA** is a significant step in the direction of being able to more fully tap this power. But much remains to be done.

Within formal education, we need to integrate our thought process into the substantive treatments that already are there. A revolution fomented from *within*, and radiated upward by students, stands a much greater chance of succeeding than a declared war waged from outside and targeted at faculty. Let us add value to the good theoretical work that already exists in fields like economics, business and sociology by incorporating "gremlins", showcasing disequilibrium phenomena and enforcing structural/conceptual rigor. The two series "Learning Laboratories In: . . ." and "Great Thinkers In: . . ." are a beginning. We are interested in seeing the two series filled out with several additional books, and have put in place a commercial apparatus to do so. To this series, other "learner controlled" laboratory environments need to be added. Be creative!

Outside of formal education, some of us should try shifting our orientations from producers to "facilitators of production". In this regard, something I call "Strategic Forums" is an activity that has been successful. In such a Forum, a group of people with a common strategic interest gather around a Macintosh equipped with **STELLA**. Following a discussion of "what's the problem behavior(s) of interest?", I facilitate peoples' efforts to make their hypotheses explicit on the screen of the Mac. A series of questions like: "What is the single most important accumulation in the system?", "What rates of flow are associated with this accumulation?", and "What resources are associated with *causing* these flows to occur?", has proven very effective in getting things rolling. Once the process is flowing, my role converts to that of a "kibbitzer" --throwing in useful suggestions from time to time, reminding the group of the "system as cause" perspective, or suggesting some "molecule of structure" that might be appropriate. It takes a certain degree of confidence to conduct such sessions. You are not "in control" to the degree that you would be if you had spent several months constructing a model, and therefore entered the situation "knowing all the answers". But then, you must ask yourself: "Is your purpose really to be 'in control', or to effect a productive change in thought process and viewpoint?" If you answered the latter, a tool now is available to help you realize your ambitions.

System dynamics is an incredibly powerful framework for learning. It truly is a tool for our times. The important problems that we will face from now on all will be "systems" problems. It therefore is time to get very serious about improving our ability to transfer our process. **STELLA** is *serious*. . . but still fun to use!