Systems Dynamics Using
Vensim Personal Learning Edition (PLE)

Download Vensim PLE at http://vensim.com/freedownload.html

Quick Start Tutorial

Preliminaries

Vensim PLE is software designed for modeling one or more quantities that change over time. Open Vensim PLE and you will see either a blank white screen or the program will attempt to load the last model that you had open. To create a new model, simply go to *File* and select *New Model*. Click *OK* to bypass the model settings window that pops up. Do not worry about changing the model settings at this point. Before jumping into a hands-on example, there are three key buttons worth taking a look at:

🔒	The lock button freezes the screen. While this button is selected you cannot change the contents of the Vensim PLE window.
🖐	The selection button allows a user to select or drag an object in the window.
⬤	Some good humor from the designers of Vensim PLE, this icon is the delete button. Select it then click on an object to delete that object.

NOTICE: Vensim PLE has an auto-save feature. If you have not saved your model after about five minutes you will be prompted to save. Saving this tutorial model is optional.

Part I – Building the Model

Before you begin building your model, take a look at the basic building blocks that you will use to create the model:
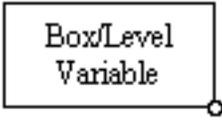
Box/Level Variable	**Box Variables** represent quantities. These are the main "nouns" in the system. Box Variables are also referred to as Level Variables.
Rate	**Rates** represent a change in a quantity over time. These are the "verbs" in the system. Rates coming out of a box variable decrease that quantity. Rates going into a box variable increase it.
Auxillary Variable	**Auxiliary Variables** can contain constants or other parameters. Auxiliary variables loosely correspond to "adjectives" and "adverbs" in the system.
(connector)	**Connectors** illustrate dependencies between variables.
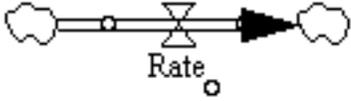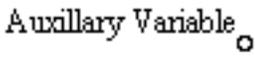
Figure 1: Basic Building Blocks

D. Hillman – National Computational Science Institute

IN this example, you will model the following situation: On a Monday 100 students return to school, one of whom has the flu. As time passes, the sick student unwittingly infects his classmates, who in turn infect others. This could be generalized as follows:

Over time, susceptible people get sick and become infected people.

To begin building the model, create two box variables: one for *Susceptible People* and one for *Infected People*. These will allow you to keep track of how many healthy and sick people there are.

PROCEDURE: Click on the Box Variable button ▨ on the toolbar. Now click in the workspace where you want to add a box variable. A blank box will appear with a blinking cursor inside. Type in a name for the box variable and press enter. By clicking again in the workspace, you can continue to add box variables until you select a different button from the toolbar.

Figure2: Adding Box Variables

NOW, use a rate to connect these two box variables. The rate represents a place in the system where something happens. Since this rate represents the number of people who change from susceptible to infected, name it *Get Sick*.

PROCEDURE: Click on the Rate button ⧖ on the toolbar. Move the mouse over *Susceptible People* and the cursor should change its icon. Click and release inside *Susceptible People*, then click and release inside *Infected People*. A new box will appear between the two box variables with a cursor inside. Type in the name of the rate and press enter. The text box is then replaced with a rate resembling the one in Figure 3.
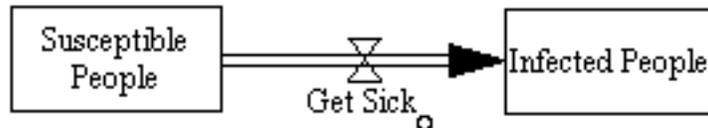
Figure 3: Adding a Rate

NOTICE: Pay attention to the direction of the arrow on the rate. The rate should go from *Susceptible People* to *Infected People*, not the other way around (at least in this model). If you see little clouds (like the rate in Figure 1) at either end of your rate, then it is not correctly connected to a variable.

Don't worry though. Click on the Delete button ◖ on the toolbar, and then click on the hourglass in the center of the rate. This erases the rate and you can try again.

D. Hillman – National Computational Science Institute

NEXT, we will need an auxiliary variable to help calculate the rate at which susceptible people become infected people. Call this variable *Infection Fraction.*   Keep in mind that this is a coefficient and not a rate.

PROCEDURE: This works just like adding a box variable. Select the Auxiliary Variable button

![VAB icon] from the toolbar. Now click in your workspace to add the variable. Type in the name and press enter.

Figure 4: Adding an Auxiliary Variable

FINALLY, Vensim PLE needs instructions about which elements depend on each other. In this case, the rate *Get Sick* depends on the number of *Susceptible People*, the number of *Infected People* and the *Infection Fraction.* Therefore, we need to add connectors from each of these variables to the *Get Sick* rate.

PROCEDURE: Click on the Connector button ![connector icon] on the top toolbar. Click and release on a variable then click and release on the object you want to connect to. Repeat this process to connect *Susceptible People*, *Infected People* and *Infection Fraction* to *Get Sick.*

NOTICE: At first, the arrows may cut through other objects or object names. You can reposition connecting arrows to make your graph more organized. To do so, click on either the Selection button

![hand icon] or the Connector button ![connector icon], then click on the small empty circle near the middle of the arrow and drag it. As you drag, the arrow will change its length and arc. Try to keep your arrows simple and direct without obscuring other objects or names.
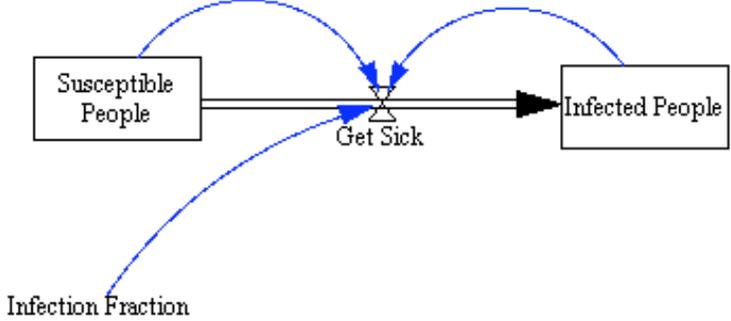
Figure 5: The Visual Model

D. Hillman – National Computational Science Institute

Part II – Specifying Input

At this point, your model should look something like Figure 5. Visually, you can see that susceptible people get sick and become infected people. The thin arching connector arrows suggest that this process is based on an infection fraction, the number of susceptible people, and the number of infected people.

The framework for the model is now in place, but you haven't given it any data yet. Before simulating the model, you must define how many susceptible people and how many infected people there are when the model starts (at time zero). You must also construct a formula for how people get sick and specify the infection fraction.

RECALL that our hypothetical school has *100* kids. Set the initial number of susceptible people to *99* and set the number of infected people to *1*.

PROCEDURE: Click on the Equations button ![Y=X²] on the toolbar. Every object that now appears in black is not sufficiently defined. Click on *Susceptible People* and a window like Figure 6 will pop up. This probably looks complex, but you won't need most of these options at first. The cursor should already be in the box to the right of the words *Init Value*. Here, type in *99*. Then click in the box to the right of *Units* and type *people*. Now, click *OK* in the bottom left of the pop-up window.

The procedure for setting the initial number of *Infected People* is analogous, except of course that there is only *1* infected person at time zero.

NOTICE: It may seem as though labeling the units is redundant here, but labeling units this way will enable you to use Vensim's Unit Checker, a very useful tool for checking for consistency in your model.
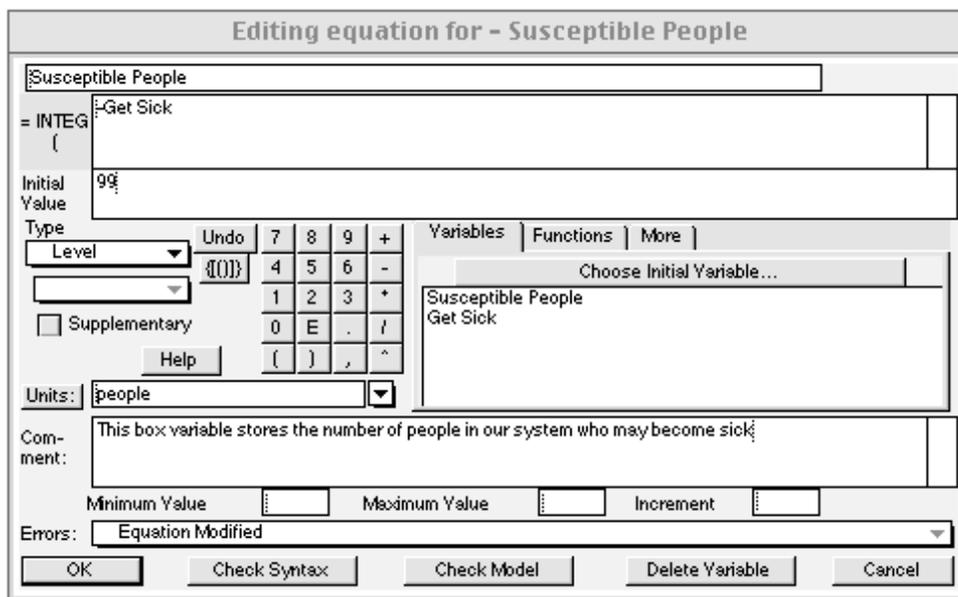
Figure 6: Editing Equation / Input Window

D. Hillman – National Computational Science Institute

NOW, we need to define a formula for the rate *Get Sick*. Assume that every susceptible person interacts with every infected person once per day. This can be expressed mathematically as *Susceptible People * Infected People*. Not every interaction results in an infection though, so multiply this total number of interactions by the infection fraction to get the number of people who get sick.

PROCEDURE: With the Equations button still selected, click on *Get Sick*. Notice that the other variables in the system are listed in the box on the right of the input window. Because you connected these variables to this rate, they must be used in the formula you build. You can add them to your formula by clicking once on them. Enter the following formula:
> *Susceptible People * Infected People * Infection Fraction*

Leaving the units blank for now, click *OK* to finish.

FINALLY, we must specify a value for *Infection Fraction*. Assuming that in two days there will be three new sick students, the infection fraction estimation is *0.0151*.

PROCEDURE: Make sure that the Equations button is still selected. Click on *Infection Fraction* and a window will appear similar to the one for box variables. Type *0.0151* in the box to the right of the equals sign near the top of the window. Then set the units to *1/(people*day)*. Click *OK* to finish.

NOTICE: For more information about how to calculate the infection fraction, consult the end of this tutorial.

Since you skipped labeling the units on *Get Sick* the first time through, you have to go back and add them in. Bring up the input window for *Get Sick* and click on the word *Units*. Vensim PLE automatically fills in these units for you based on the equation for that rate.

That takes care of defining the input. If you need to change your initial values or the formulas later, just select the equations button and click on the variable or rate that you want to edit.

Part III – Running the Model

Before simulating the model with the run button ![run button], there are a few things left to specify.

Model Settings

Click on *Model* from the menu at the top and select *Settings*. A menu resembling Figure 7 will appear. This menu addresses some of the fundamentals of computational science. You need to specify the start time, the stop time, the time-step, and the units for time. Run the model for ten days. Set *Initial Time* to *0*, *Final Time* to *10* and *Time Step* to *0.1*. To match the *Infection Fraction* units, select *Day* from the drop-down menu next to *Units for Time*. Click *OK* to finish.
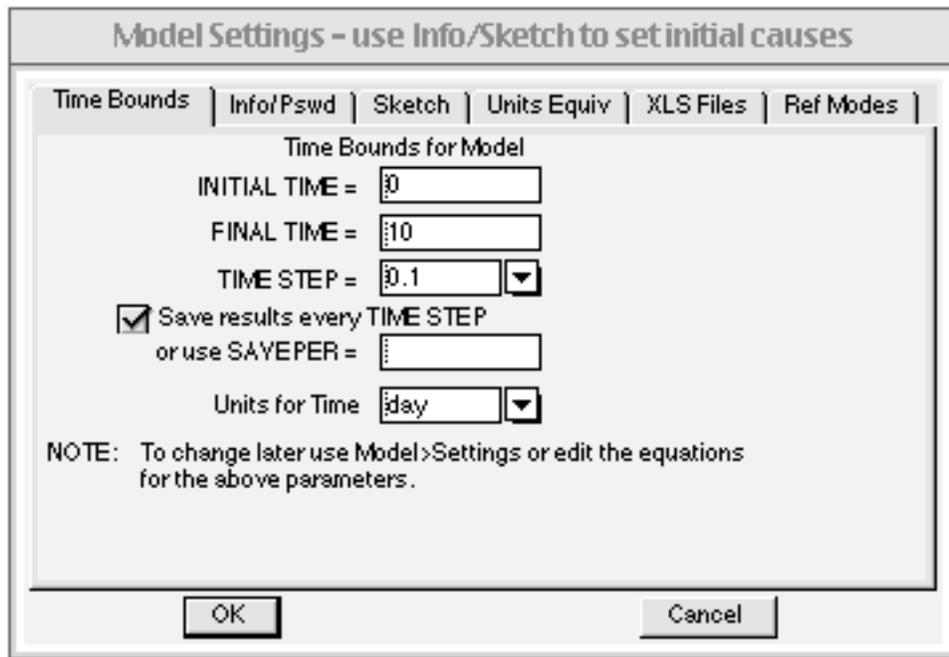
Figure 7: Model Settings Window

Checking Units

There is one last thing to do before running the model. On the top menu, select *Model* and then click on *Units Check*. Vensim PLE will check for any inconsistencies in units throughout the model. If all is well, you get an *A O. K.* message. If not, a new window will explain where inconsistencies were found.

Now you are ready to click Run ![run button] and simulate your model. You may be asked if you wish to save over the current run. Click *yes*. Though it may appear that nothing has happened, as long as you didn't get an error message, your model was simulated.

D. Hillman – National Computational Science Institute

Part IV – Examining Output
There is no default output for data generated in Vensim PLE simulations. Output can be viewed on quick graphs/tables, custom graphs/tables and interactively with *SyntheSim*.

Quick Graphs and Tables

To view a graph or table of a single variable over time, choose the Select button and click on the variable in question. On the left-hand toolbar, you should see the icons in the chart below. These tools are designed for quick viewing of variable values:

	Causes Strip: This brings up a strip graph for the selected variable and any variables affecting it in a causal loop.
	Graph: This will bring up a graph of the selected variable over time.
	Table: Displays values for the selected variable horizontally.
	Table Time Down: Displays values for the selected variable over time vertically.

Figure 8: Quick Output Buttons

These buttons offer immediate access to data but are somewhat limited. You can only include one variable (other than *Time* itself) at a time on a graph or table, and the graph disappears when you click outside of its windows boundaries.

Custom Graphs and Tables
Vensim PLE's real output capabilities are utilized through custom graphs and tables. There are many more possibilities than can be addressed here, but this guide will walk through simultaneously looking at susceptible and infected people over time in both graph and table forms. Consult Figures 9-11 on the following page while working through this procedure.

PROCEDURE: First, open the Control Panel by clicking on the button. This brings up a new window. Click on the *Graphs* tab and then click the *New* button for a new graph. Name and title your graph something like "Population Status" and "Susceptible and Infected People over Time." Click on the box labeled *Sel* to the right of *X-Axis* and choose *Time* from the pop-up window.

The rows in the bottom right-hand corner are for adding variables to the Y-axis. Click on the first and second "Sel" buttons and choose susceptible people and infected people respectively. If you click *OK*, the graph will appear in the Control Panel window under the name that you gave it. To view your graph at any time, open the Control Panel and double-click on the graph name or select the name and click *Display*.

You may receive an error message that says, "Graph requires datasets to be loaded." In this case, run your model again before trying to view your custom graph.

To create a table, repeat the process but stop short of clicking *OK* and instead select *As Table…* The window will change, offering new options for formatting a table, but you can access the data by simply clicking *OK*.

D. Hillman – National Computational Science Institute
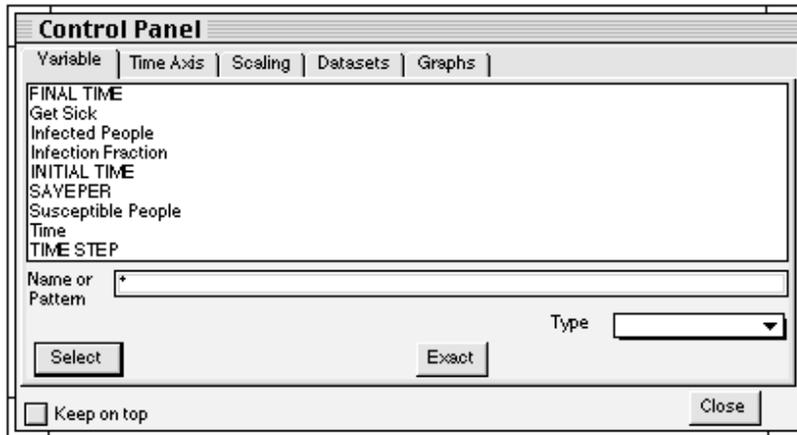
Figure 9: Control Panel Window
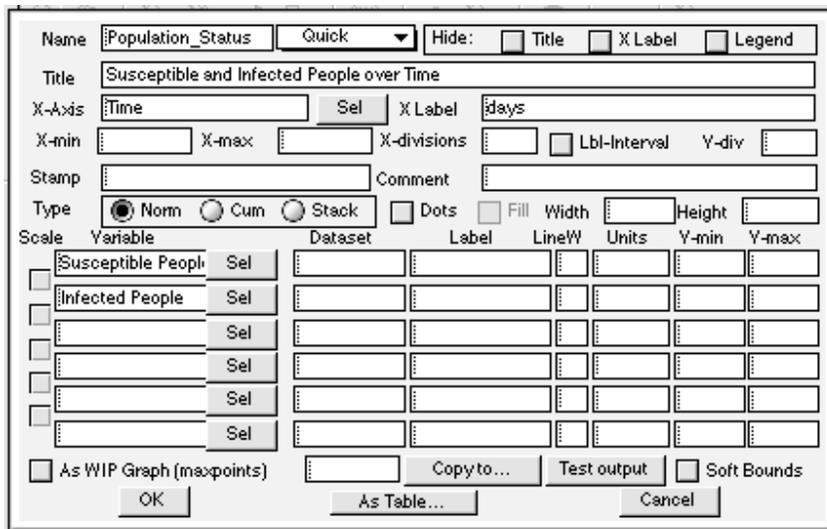
Figure 10: Custom Graph Design Window
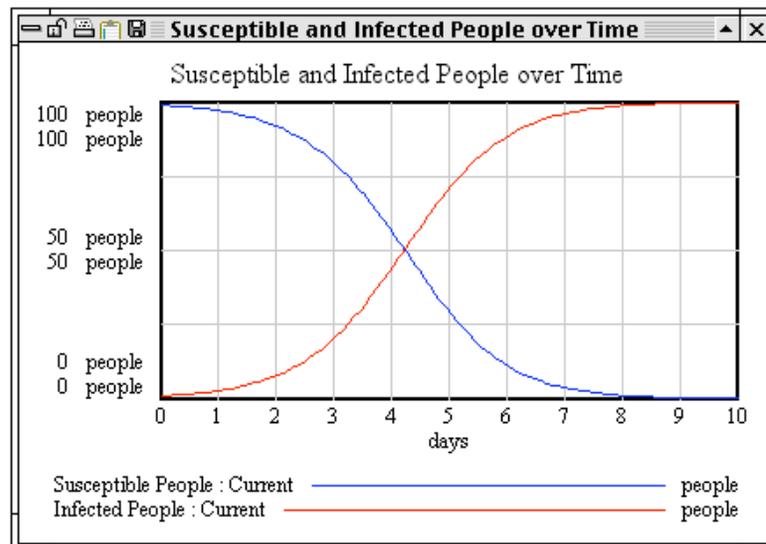
Figure 11: Custom Graph of Population Status over Time

D. Hillman – National Computational Science Institute

SyntheSim

Next to the standard run button [icon] is the *SyntheSim* button [icon]. *SyntheSim* can add considerable interactivity to a model. Clicking this button turns all constant-valued auxiliary variables into slider bars. It also places small graphs inside the boxes containing the names of box variables and rates. When *SyntheSim* is on, any change you make will be immediately reflected in the graphs on each box variable and rate.

> **PROCEDURE:** Values of constant auxiliary variables can be adjusted by dragging the slider bars or by clicking them. If you click on the arrows on the sides of the bars, a pop-up window will appear. In this window, you can manually adjust the value, or set the minimum, maximum and increment for that slider bar.

[icon]	Exits *SyntheSim* mode
[icon]	Saves the current settings and data
[icon]	Resets the slider to the value of the related variable when *SyntheSim* was started
[icon]	Resets all of the sliders

Figure 12: SyntheSim Toolbar Buttons
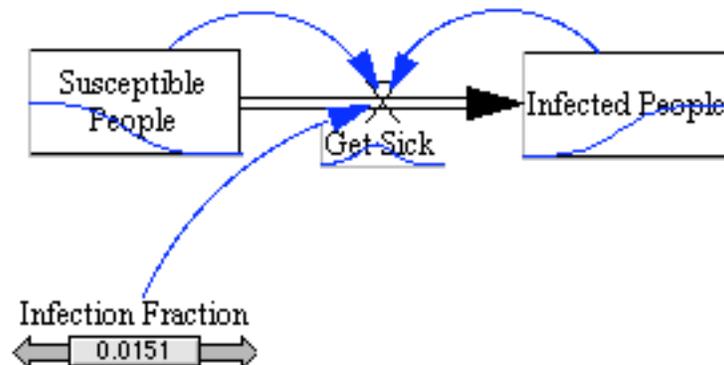
Figure 13: SyntheSim Mode

D. Hillman – National Computational Science Institute

Part V – Interpretation and Analysis

Based on the custom graphs you generated, what can you conclude? Around the start of the third day of the simulation, there is an even number of susceptible and infected people. In addition, this is the point at which the disease peaks in terms of the number of people getting sick. To see this make a quick graph of *Get Sick*. Select *Get Sick* with the selection button

, and then click the graph button   to see the value of *Get Sick* over time. In the long term, what happens in this model? Everyone gets sick and stays that way[1]. This may not be entirely realistic, but it is in fact what we told the computer to model. Take some time and list other assumptions made by this model.

It is worth noting that even the most sophisticated models involve assumptions and simplifications. The real question is whether or not the assumptions are reasonable and yield useful information. We could have made our model more accurate (for example, by including a box variable for recovered people, or looking up rather than estimating the infection fraction), but that would necessitate making the model more complex. Scientific modeling requires a constant give-and-take between accuracy and complexity.

As a final word on this model, note that the assumptions made in it should be viewed as opportunities for exploration rather than limitations. Challenge yourself and/or your students to expand the model. What if infected people can get well? What if people who get well can get sick again? What if some people who get well develop an immunity to the disease? Visit the Center for Disease Control web site and search for real world data. Part of the appeal of interactive tools is that models you build, even simple ones, can help pose and answer numerous questions. Expanding and refining models builds pattern recognition and offers insight into the way systems and numbers behave.

---

[1] Actually, the quantity *Infected People* approaches 100 but never reaches it, because of the nature of the numerical model.

D. Hillman – National Computational Science Institute

Estimating the Infection Fraction

When defining the input for this sample model, you might have wondered, "How did someone come up with the value *0.0151* for the infection fraction?"

First, let's agree on some abbreviations for our variables. Let *S* be the number of susceptible people and *I* be the number of infected people. Call the infection fraction *k*. We will also abbreviate the rate Get Sick as *GS*. Recall that the formula for Get Sick is

$$GS = S * I * k$$

Since *GS* is a rate, it represents a change over time. In other words, the change in the number of infected people per unit of time (*GS*) is equal to the number of susceptible people times the number of infected people times the infection fraction. Using notation from Calculus, this can be written:

$$dI/dt = S * I * k$$

dI means "the change in *I*" and *dt* means "the change in *t*."

Finding values for *S* and *I* is not too hard, but what *k* should be is less obvious. One way to calculate a reasonable value of k is to estimate it based on personal experience. If one sick person comes to school, how many students would you expect to get sick each day?

Suppose you decided that in two days, three new students would be sick. We can now substitute *3* for *dI* and *2* for *dt* in the equation above to get
$$3/2 = S * I * k$$

We also know that there are *99* susceptible people and *1* infected person at time zero, So:
$$3/2 = 99 * 1 * k$$

Now, solving for *k* indicates what the infection fraction should be so that in two days there will be three new sick students.
$$k = (3/2) / (99 * 1) = 1.5 / 99 = 0.01515…$$

This is a repeating decimal, but rounding off to *0.0151* is more than generous if you consider significant figures.

If you plan to go through this calculation in a classroom setting, consider polling students about how long it would take for people in the model to get sick, averaging their responses, and using that estimate to calculate *k*.

D. Hillman – National Computational Science Institute

More Key Points about Vensim

COST: It's FREE! Unlike similar products such as Berkeley-Madonna and STELLA, Vensim PLE is available for educational and personal (non-commercial) at no cost. To obtain Vensim PLE, visit their website at http://vensim.com/freedownload.html. While individual and site license discounts may be negotiated and result in a lower cost, Berkeley-Madonna typically costs $299 for a single license, (http://berkeleymadonna.com/pricing.html). Similarly, without a special agreement, Stella costs $649 dollars for a college faculty license and $199 for a K-12 teacher's license (www.iseesystems.com). Almost anything you can do with one of the tools can be recreated with the others, and because cost is a consideration for every educator, Vensim PLE is a very appealing option for aggregate and systems modeling.

GRAPHICS AND VISUALS: Though the graphics of Vensim PLE are not visually striking at first, the possibilities for customization are manifold. Control-clicking or right-clicking on an object brings up a menu with choices for color, shape, font and size. The toolbar at the bottom of the screen offers even more visual flexibility. With a little practice, you can take control of your model's appearance.

DIFFERENTIAL EQUATIONS: The underlying differential equations can be accessed by clicking on the Doc button Doc on the toolbar to the left.

INTEGRATION METHODS: The two integration methods offered in Vensim PLE are Euler and Runge-Kutta 4. To select an integration method, click on the Set button SET, and then click on the integration method to the left to change integration methods.

INTERFACE LEVELS: Although Vensim PLE does not have layered windows in the same sense as STELLA interface, model and equation levels, it does offer extendable window space and multiple views. Controls for the multiple views are in the lower left of the screen.

FILE EXTENSION: Vensim PLE model files are typically saved with the extension .mdl. These files cross platforms flawlessly between Windows and Mac machines. Also, you can open these files with a text editor and edit them manually! Information from specific runs is commonly saved in .vdf files.

AUTO-SAVE: As mentioned, Vensim PLE will prompt you after five minutes whether you wish to save any unsaved model, so that it can begin auto-saving a backup. This can be very useful in preventing loss of work and/or data. Back-up models are saved as *Filename.2mdl*

DOCUMENTATION: You can download the Vensim User's guide by going to http://www.vensim.com/documentation.html and following the link to "User's Guide in Adobe Acrobat as a zip compressed file."

This guide is an adaptation of *Modeling Change Using Stella* by Dr. Holly Hirst of the National Computational Science Institute