

Guidelines for third Breakout Group presentation: Dynamic Hypotheses¹

1. Purpose of third client-week and breakout presentation: Dynamic Hypotheses.

In the third week you will develop – and then present in your breakout group – a set of causal loops that you and your client believe can generate the patterns of behavior captured in your reference modes. The combination of the loop diagrams and the reference modes *are* your dynamic hypotheses, theories that some particular structure *can* generate some particular pattern of behavior.

Very important is to organize the dynamic hypotheses (causal loops + reference modes) in a manner that serves three purposes. The hypotheses should be organized in a way that...

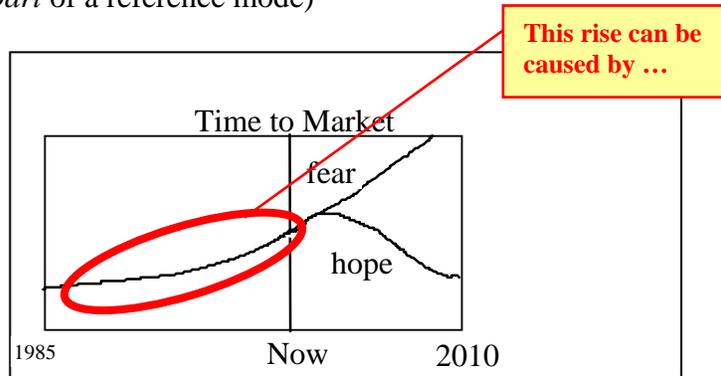
- Allows *your client* to explain to her colleagues *all* of the causal processes that she believes can generate the behavior patterns – good and bad – that characterize the problem.
- Constitutes a “road map” for your simulation modeling activity that will be beginning soon. That is the hypotheses are an implicit contract between you and your client that you will model one hypothesis to begin with, completely analyze it, and then go on to another hypothesis. You will stop modeling when you’ve either (a) gone through all of the loops (unlikely given the time constraints of this course), (b) reached the limit of your own or your client’s desire to continue (also unlikely), or (c) run out of time (most likely).
- Fosters loop-based policy analysis. You will want to consider the policy implications of all of the dynamic hypotheses, even though you probably won’t have time to add all of them to your simulation model(s).

As before, please hand in hard copies of your overheads.

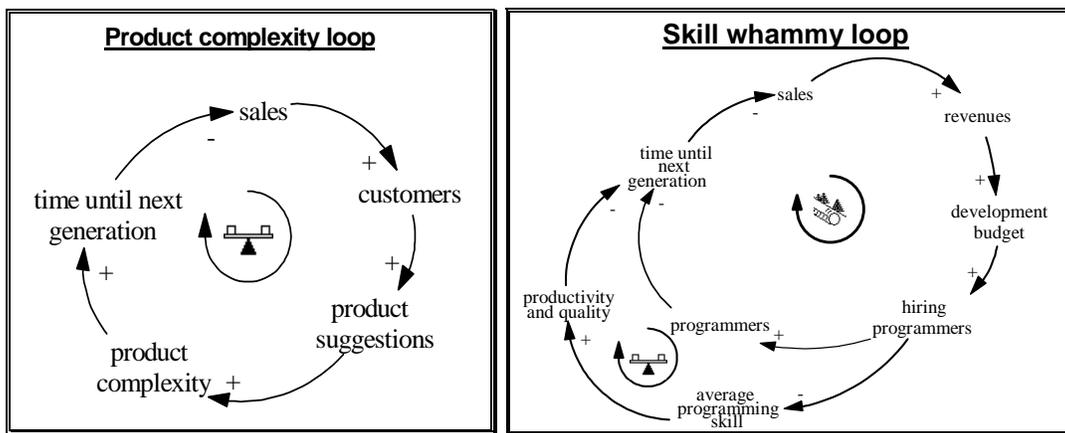
- 2. Organizing the hypotheses.** A diagram that simply showed all loops would look like a plate of spaghetti, and would be incomprehensible to even the most experienced SDer. You need to figure out a way of presenting the loops so that your audience (or your client’s colleagues) can see hypothesis separately. There are two strategies: (1) create a separate diagram for each hypothesis and link them together verbally in the presentation, or (2) create a layered causal loop diagram.

¹ Prepared by Jim Hines 1998. Revised February 1999. Revised June 1999. Revised February 2000. Revised February 2004. Copyright © 1998, 1999, 2000, 2004 Jim Hines.

The first approach is straight forward. For example, say you have two explanations for the increasing part of the following reference mode (note a dynamic hypothesis does not have to involve the *entire* reference mode). As in this example it may involve only *part* of a reference mode)

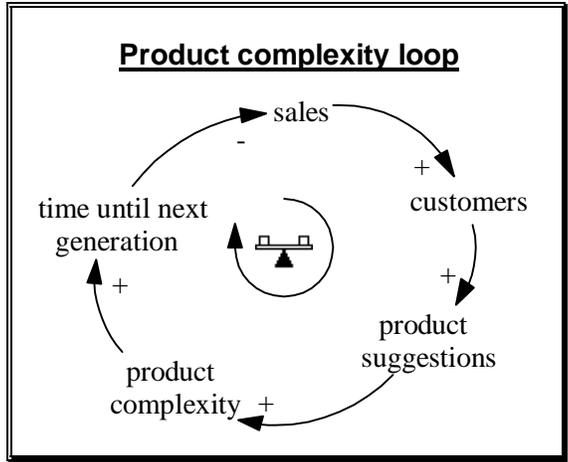


First more customers provide more product suggestions and, hence, lead to a more complicated next product generation. Second, as the company's sales grow it can afford more programmers, but the additional programmers' inexperience slows down the process. You could simply create two pictures:

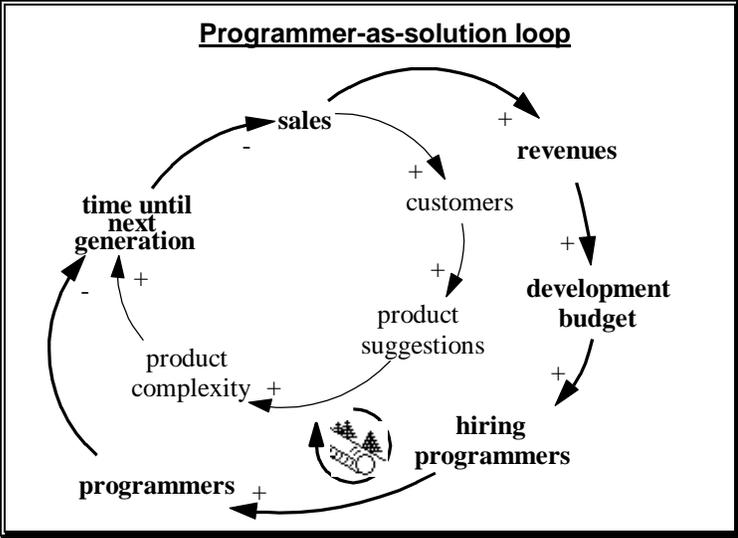


One strategy for dealing with this potential confusion is to present one or two loops at a time, with each new loop layered on top of what has gone before.² In this case you might start with a loopset representing the first structure.

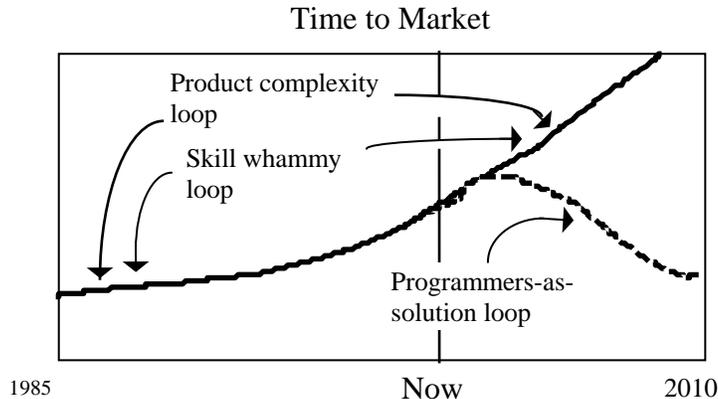
² An alternative approach is to develop a *separate* causal loop diagram for each of your hypotheses. An advantage of this approach is that it may keep your loops clearer. Two disadvantages are that (a) you will not see the inter-relationships between the loops as well, and for that reason (b) it is less likely that you will discover new loops by looking at your diagram. In some cases, and perhaps in many cases, the advantage outweighs the disadvantage. Please feel free to pursue this alternative if you wish. Tell your breakout group how this different approach worked (or failed to work) for you.



Then, you would layer on the second loopset in one or two steps. For example in two steps:



3. **Causal loops linked to the dynamic hypotheses.** As you move through your loops, you will want to indicate what behavior each loop or loopset contributes to. One way to show the connection between loops and behavior is to “decorate” your reference modes with the appropriate loop names. For example,



Another way of linking loops to behavior is shown in the attached Nynex slides.

4. **Thought Process.** Coming up with causal loops is not always easy. A few hints might be in order.
- a) Start by simply putting down a variable that relates to your (at the moment, ill-formed) hypothesis. Then move backwards causally. Ask yourself what causes the first variable; put down the cause as a new variable, and draw the link. Then what causes this new second variable? As you draw, keep curving the arrows so that the string of causal variables begins to form a loop. Eventually, you will come (one hopes) to a causal variable, which is itself caused by the first variable you put down. Voila! You have a loop.
 - b) Undesirable and positive loops are often easier to find than either desired loops or negative loops. Ask the question “What makes this mode grow?” Or “what keeps this mode from growing”.
 - c) Hypotheses that involve solving a problem or moving toward a goal are usually negative loops. (E.g. “we’ll lower prices to boost our market share” is a negative loop whose “goal” is to bring actual market share up to desired.)
 - d) You can often find a “counter-acting” loop. If you have a desirable positive loop (e.g. programmers-as-solution) look for an undesirable negative loop (e.g. skill-whammy). And, if you have an undesirable positive loop, look for the desired negative loop. And vice versa. (These last two points amount to completing a *limits to growth* archetype and a *fixes that fails* archetype respectively).
 - e) You do not have to show all the loops, but only those that might be driving the reference modes. That is, only the ones that relate to your hypotheses about what structure could cause what behavior.
5. **Client Process.** The client process also requires some attention. Usually loops can be developed with the client in some manner. Layering them or arranging individual loops in a coherent sequence, though, is usually done off-line.

Expert modelers commonly “do” looping live with a client group. But doing so takes experience and confidence in the process, two items which may be in short supply when you are doing your first few projects. Fortunately, interviews also work well here. The technique of “taking over a conference room” and having interviewees cycle through is quite doable. Cover a wall with sheets of flip-chart paper and draw a big diagram. Schedule time between interviewees (e.g. half an hour) so that you have time to think about and draw (or redraw) loops that emerged (perhaps not explicitly) during the prior interview, so that you can discuss them with the next interviewee. A similar procedure using telephones or videoconferences can be developed for long-distance clients.

You have a headstart, of course, because last week you developed *rough* hypotheses. Be careful, though: Drawing loops in advance and *presenting* them for a critique to your client will likely create undesirable roles for you and your client. Your client naturally will tend to assume the role of skeptic, naysayer, and faultfinder; while you will tend to assume the role of loop-defender. It’s always possible to nit-pick loops, so you will find yourself in a losing position. Your client will feel the process is not working, because he’s able to argue against the loops, and both you and your client will feel that that the “other side” is being unreasonable and uncooperative. If you must create loops in advance, keep them to yourself; then “pull them out of your client” – perhaps in response to verbal comments he or she makes that seem to hearken back to your loops. The feeling that you engender in your client should be that the loops are giving form to his or her ideas. You and your client should feel that you are sitting on the same side of the table.

Eventually, as more loops get put on the growing diagram, your client will start adding links. At this point it is possible to lose control as connections between variables are drawn helter-skelter, without regard for whether the latest connection is relevant for an important loop that relates to the reference modes. Try to describe a new loop that is created by each new link. To the extent possible hook the loops to the reference modes as they emerge.

But, truthfully it’s difficult not to lose some control here, particularly if you succeed in igniting the enthusiasm of your client. A good place to regain your balance is in the offline process when you either layer your loops or arrange the individual hypotheses into some meaningful order...

6. **Layering and Arranging.** To create a layered diagram you should use either Vensim DSS’ view mechanism or Vensim’s “levels-of-hiding” feature. The advantage of using the view mechanism is that you change the line thickness, font, or color for a loop from one view to the next. With the levels-of-hiding feature each loop is either shown or not shown – you don’t have the ability to show it but with a different style.

To use views, first draw the complete diagram in Vensim, and copy the complete diagram onto lots of views. Then go to the first view and, using the magic wand, hide

everything except your first loop. Now go to the second diagram and hide everything but your first and second loops. Etc.

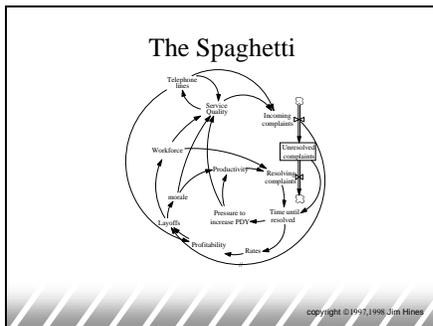
To use the levels-of-hiding feature, work backwards: Start at the most hidden level and hide everything except the last loop you intend to uncover. Then move up a level and hide the second to last loop you intend to uncover. Move up and hide the third-to-last loop, etc. If this is too hard mentally, you can work forwards by first hiding everything except your first loop, then moving down a level and hiding everything but your first and second loop, then moving down again and hiding everything except your first three loops, etc.

Whether you are using the layering approach or the individual-hypothesis approach, you ought to consider the order in which you will show the loops. You may decide to choose a story-telling or chronological order – (e.g. “this positive loop was causing increasing complexity, but it was perceived as increasing development time and so management mandated training from all programmers. The long term impact of course would be to shorten development time, but the immediate impact was to further lengthen development time as developers were taken out of production and put into the classroom (show unintended positive loop)). Or, you might want to take a logical approach – (e.g. “the negative loop is destabilized by the following two positive loops”). Or some combination of these may serve you best. If you are at a loss, you may simply want to consider how to transition from one loop (or hypothesis) to the next without worrying too much about the coherence of the different transitions.

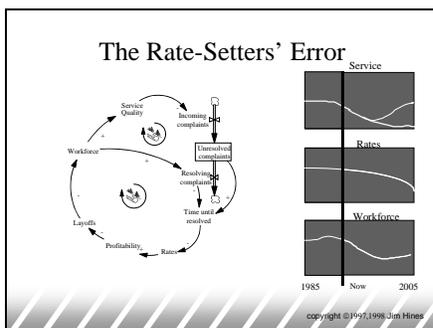
7. **Review with the client.** Once you have everything organized, review it with your client as soon as you can. If possible, do it this week, if not do it next week. Either way, the meeting where you go back over the hypotheses with your client will be extremely rich for your client. You may want to schedule two hours or more for this meeting (more than four hours is probably not necessary). During this meeting work with your client to draw out insights and policies.
8. **Insights and policies.** Whether or not you’ve had a chance to go over your organizing/layering work with your client, *you* will have insights from doing the looping and organizing the loops. The dynamic hypotheses will themselves represent insights and inspire policy ideas. You also may find additional insights concerning the standard method or some aspect of the client’s situation. Please prepare a fast slide of insights and policies that emerged this week. (In the following presentation (#4) you will be asked to dwell a bit more on the impact of your work thus far).

Example slides

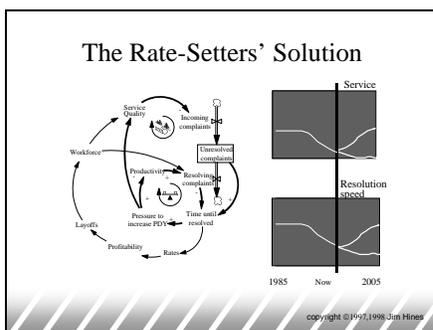
Slide 1



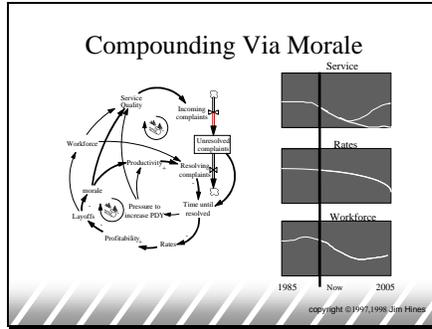
Slide 2



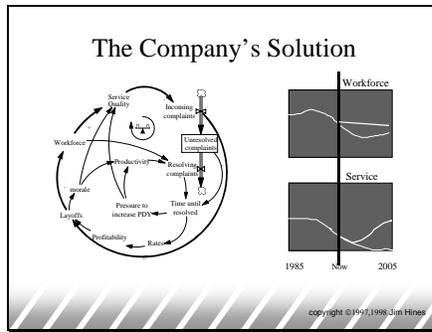
Slide 3



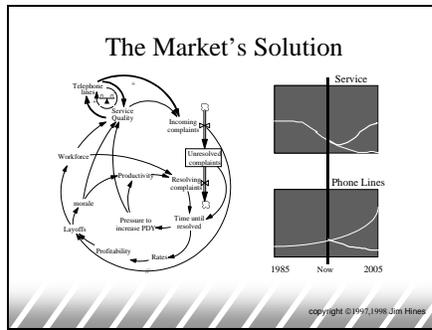
Slide 4



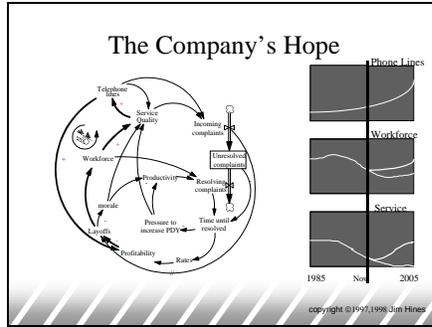
Slide 5



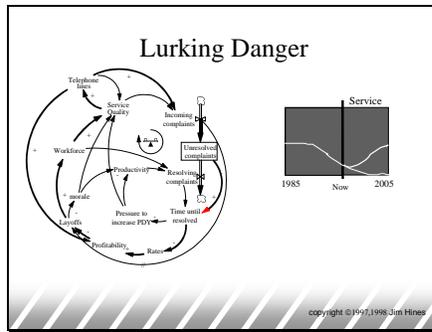
Slide 6



Slide 7



Slide 8



Slide 9

- ### Insights and Policies (tentative)
- ◆ The rate setters' error
 - Reverse current "incentive"?
 - Cut different workers?
 - Re-categorize complaints?
 - ◆ The rate setters' solution
 - Respond to rate threat by improving quality
 - Resist pressure on productivity
- copyright © 1997, 1998 Jim Hines

Slide 10

Insights and Policies (tentative)
continued

- ◆ The Market's solution
 - Monitor more than just complaints or service!
 - Put in place a system for spotting a situation of slowing (declining) growth and increasing service

copyright © 1997, 1998 Jim Hines