“turning data into dollars”

Tom’s Ten Data Tips – August 2007

## System Dynamics

System dynamics are characterized by the fact that adding up intentions and actions of the constituent parts is not enough to explain an entire system’s behavior. Such systems are also often called non-linear, complex systems and one of their characteristics is extreme sensitivity to initial conditions. The metaphorical example is a butterfly in India flapping its wings that precipitates a hurricane in America.

Sensitivity to initial conditions implies that a minute change in the initial state of the system, can result in erratic and unpredictable behavior of the system. This occurs despite the fact that the behavior may be governed by well-defined, completely deterministic mathematical rules. System dynamics are a language to represent such complexity.

Characteristics of complex systems are:

- Tight coupling – everything influences everything else in some (indirect) way
- Dynamic – change keeps occurring, often on multiple time scales
- Counterintuitive – because cause and effect are distant in time and space, it is not always obvious where high leverage points of the system are
- Exhibits tradeoffs – often long-run behavior is opposite to short-term effects. E.g., in many cases, things need to get worse before they can get better

## 1. Linear Thinking Is A Dangerous Phallacy

From early on in our education, we are taught to immediately solve problems that we face. This tendency is so deeply ingrained, that we are often hardly aware of it. It has two dreadful side effects. One was discussed in an earlier newsletter “problem analysis” – tip# 1: The #1 Threat To Accurate Problem Definition Is The Irresistible Urge To Solve It, and another one is referred to as “open loop thinking.”

“Open loop thinking” is the sequence from: problem identification – data gathering – evaluating alternatives – selecting solutions – implementation. It’s called an “open loop” because the consequences of implementation on creating new problems are often ignored. In hindsight, the undesired side effects following implementation are blamed for the problems when there really is no such thing as side effects, there are only *effects*. They're called "side-effects" because they are an unwelcome (and often surprising) consequence of our own implementation.

## 2. Cause And Effect Are Distant In Time And Space In Complex Systems

In "real life" our decisions and actions often have multiple consequences. For example, you lower the price on a product to accelerate sales. Initially this seems to work. In the short term, sales indeed increase. But as a "side effect", the quality perception of the product also might go down, decreasing the intrinsic attractiveness of the product.

Because the long-term effect on sales operates with a delay (namely via the lowered quality perception of the product) this serves to reinforce the *immediate* connection between lowering the price and accelerating sales. At the same time, it is almost as if the effect of a price cut on quality perception is "invisible", because the connection is distant in time and place.

## 3. Non-linear Behavior (Generally) Has *No* Analytical Solution

Apart from the most trivial first order feedback loops, complex systems can not be solved analytically. This "problem" has been largely responsible for the neglect of this field. However, systems that have no analytical solution, can nonetheless be simulated, and can very well be subject to experimentation. This is where the power of computers comes in.

It is only in recent times that practical software solutions have become available (e.g. Vensim, Stella, PowerSim, etc.) to model complex systems. This allows for so-called brute force experimentation. Although the mathematical elegance may be lacking, this is nonetheless preferable over making simplifying assumptions in order to reduce the system to a form that *is* analytically solvable (as is common practice). The tendency to treat systems as if they were linear has hampered the development of more realistic models of reality.

 

## 4. Management Flight Simulators Provide Unique Learning Opportunities

The business world leaves only limited room for experimentation. Either a policy works, or it doesn't. In the latter case, there are not many second chances, because unsuccessful managers are usually quickly taken out. Or even worse, an entire business may need to shut down.

The power of management flight simulators (simulations of business conditions) lies in compressing space and time, and giving managers an opportunity to test new policies without running the risk of burning up corporate capital. This enhances learning and insight of actual business conditions, and gives management a chance to see how business practices hold up in the face of rare or unexpected business scenarios.

Some of the most successful Fortune 500 companies (e.g. Shell, Dow Chemical, etc.) owe their current status in large part to developing robust strategies by using management flight simulators, strategies that will hold up under a variety of ensuing conditions. In times of market turmoil these corporations managed to gain relative market share.

## 5. Attempts To Solve A Problem, Sometimes Makes It Worse

One of the deep insights that system dynamics can give is that the reasons why well intentioned measures often do not work are sometimes embedded in the very nature of the system in which one operates. In project work for instance, it has been demonstrated very often that measures to catch up when the delivery schedule is falling behind often turn out to be counter productive.

How can this be? For example: as soon as a project starts to fall behind on schedule, overtime is usually the first thing management resorts to. When the schedule later falls desperately behind, new (and/or additional) hires are brought in. It turns out, this is exactly the reverse from a constructive policy. This tactic *worsens* the problem. Because new staff need to be brought up to speed (and initially distract existing project staff), their effectiveness is low later on in the schedule, and high early in the project. Overtime eventually burns up staff and induces errors, which is detrimental early on, but much more bearable later in the project schedule.

## 6. Mental Models Are An Integral Part Of System Dynamics

System dynamics are both a formal modeling system as well as a vehicle for understanding (see also tip #9). One of the most powerful tools is "mental models", the insight that our interpretation of the truth is responsible for shaping what we see and can't see.

For example, decades before the hole in the ozone layer was "discovered". It had been known to be there. However, researcher were convinced that the aberrant recordings must be due to measurement error, so the results were discarded. Parents can be oblivious to their children's misbehavior, because they tend to interpret reality different from their surrounding social world. By making explicit how our colored sunglasses taint reality, it becomes possible to challenge one's deeply held beliefs and expand the boundaries of our own mental models.

## 7. You Can Forecast Where You Cannot Affect, And Vice Versa

Complex systems display some fascinating characteristics. For one thing, the region in the future where you can forecast the systems behavior, is typically where you cannot exert any (meaningful) influence on the systems behavior. On the other hand, further out in to the future where you can influence the systems behavior, you can't make a reliable forecast. But you *can* install policies that will influence the *nature* of the systems behavior further out in the future.

## 8. Systems Dynamics Models Can *Not* Be Validated

Validation in scientific terms carries a very specific connotation. It implies verification of truthfulness by some external yardstick. In *this* particular sense, system dynamics models can *never* be validated.

One of the reasons for this, is that there is always a conceivable set of model parameters that can generate a system's behavior that equals real life phenomena to some arbitrary level of accuracy. In plain English: a system dynamics model can be made to fit to *any* set of data. A system dynamics model can be tested for truthfulness to existing data, and plausibility of effects.

## 9. System Dynamics Is *Both* A Language For Understanding *As Well* As Description

There are two sides to this coin. System dynamics is both a language to describe and formalize system behavior, as well as a tool for creating insight. Peter Senge's "Fifth Discipline" is the best known example of the latter.

As a language, system dynamics offers a set of "archetypes" that capture (almost) any system so far conceived. It is commonly believed that about one dozen building blocks are sufficient to describe any system.

As a tool for creating insight, system dynamics offers the potential to simplify an overwhelming complexity surrounding us, yet at the same time surface essential characteristic behaviors.

## 10. To Create Improvement, Decide What Should Get Worse

It is tempting to strive for universal improvement. However, this is impossible. Any change involves a trade-off between elements that should improve against elements getting worse. For instance, in city planning, if "everything" would get better, increased attraction would lead to higher real estate demand, increased population (more crowds), more traffic, etc. One would need to decide that, for instance, higher real estate prices and more traffic congestion are in order, to "justify" better education, employment opportunities, etc.

A system as a whole simply can not get universally better, because if you improve all elements, the whole *must* get worse. Instead, you decide what you want to make worse, and the rest will get better. In a supermarket, better prices, an expanded assortment, fewer stock outs, etc. will lead to longer lines. If you attempt to make lines shorter, you need more cashiers which will decrease profitability… There is always *some* tradeoff involved.