The Origin of the Goal Tree

by H. William Dettmer

"Once upon a time..."

Don't all old-time stories start like that? Well, this one does.

Once upon a time, back in late 1992, I learned about a training course in New Haven, Connecticut, called the Jonah Program. I was an adjunct faculty member at the University of Southern California (USC) at the time, teaching systems thinking and systems management to graduate students—working professionals.

A Goldratt Institute certified associate in Los Angeles recounted for my boss at USC, the director of the institute in which I was teaching, Goldratt's background and the gestation of the Theory of Constraints. In his infinite wisdom (from my perspective!), my boss decided to send me to the course. At the time, Goldratt was offering concessionary rates on tuition for his courses as a way of spreading the word through academia.

I went for my two-week Connecticut sojourn in February 1993, expecting to learn all about drum-buffer-rope (D-B-R) production control. Imagine my surprise when I got to 442 Orange Street and heard the instructor say, "Welcome to the Thinking Processes." *Not* what I'd bargained for.

But I set aside my doubts and waited to see what would happen. This was only the second Jonah course that taught the Thinking Processes (TP) exclusively. Prior to the first one, the preceding November, it had always been a drum-buffer-rope course. And that's what I'd signed up expecting. But over the next ten days, I discovered something infinitely more valuable than the D-B-R course.

It was clear to me early on that the Thinking Processes offered far more value to me, as a systems management practitioner, than D-B-R ever could. A kind of universal problem-solving tool, the TP were not limited to production systems. They could be equally well applied to any kind of system, anywhere.

"Ready! ... Fire! ... Aim!

I left the Jonah course in New Haven "with my hair on fire." As I flew back to California, I remember looking at all the passengers around me and thinking, "I know a secret that you don't know ... and it could change the world!" (Okay, I was more than a little starry-eyed at that time. But I was also unaware that I'd come crashing back to earth soon enough.)

I returned to my teaching duties at USC just in time to start a new graduate course in project management at Edwards Air Force Base, California. I resolved to start introducing the TP into my regular curriculum courses, and this would be my first opportunity.

And I fell flat on my face. The students, God bless 'em, were more patient with me than I had a right to expect. They listened with rapt attention—and quizzical looks on their faces—as I tried to explain how to build a Current Reality Tree. These were all working professionals, many of them engineers, looking to broaden their horizons and enhance their careers with a Master of Science degree in systems management. And they tried. They really did.

I was quick to recognize the reason for my failure—I had no teaching materials worthy of the name to hand out to the students. All I had was a binder full of logic trees and a few checklists that I'd been given at the Goldratt Institute. But whereas the "brute force" approach to learning the TP had the luxury of time to work at 442 Orange Street, at USC I usually had no

more than 30 minutes of a two-and-a-half hour class meeting to introduce anything new. I needed the rest of the time to cover the sanctioned project management course material. (Harold Kerzner's biblical tome was the required reading for that, if any of you project management mavens remember it.) For me, beginning to teach the TP, even to an intelligent, receptive audience, was very much a matter of "Ready! ... Fire! ... Aim!"

To the rescue...!

In disillusion, I called Bob Fox, then the North American president of the Goldratt Institute. I cried on his shoulder (figuratively speaking), recounting my tale of woe. Bob said, "I think you need to come back for the Jonah's Jonah course." *What was that?* I wondered. "It's the course to teach you how to teach the Thinking Processes," Bob replied.

So, four months later, I found myself once again at the Goldratt Institute, sitting in that marvelous Victorian conference room with the high ceilings and tall windows, and the table big enough to land an airplane on. Over the next *three* weeks (one week added, this time), Alex Klarman and Tracey Burton ran me and the other eight participants through the train-the-trainer course.

Sitting next to me were two other aspiring TP trainers: Craig Morgan, a publishing executive from Chicago, and Dr. Van Gray, a business school professor from Baylor University. They were pretty laid back. I had my nose buried in my notes, which I scribbled intensively throughout the first two weeks of the program. At one point, Craig Morgan said, "What are you doing? Writing a book?" I had some 80 pages of hand-written notes and diagrams ... added to the 60 I'd accumulated in the original Jonah course. I thought for a moment and replied, "Yes, I guess I am." And that was the beginning of my first book on logic trees.

It took me another year to consolidate all my notes into an inch-thick softcover "course reader" printed by the USC bookstore in Los Angeles, specifically to support my graduate courses. I was now ready to really teach the TP to my students! And this time, the students ate it up. Some did exceedingly well, others not so well, the majority somewhere in the middle. But my objective was achieved: All of them were enthusiastic about using the tools.

Flounder: More Than Just a Fish

Getting to the point of actually being able to teach the Thinking Processes effectively seemed a lot like cutting and slashing my way through the jungles of Darkest Africa ("Doctor Livingston, I presume...?") My students floundered with the construction of Current Reality Trees. For that matter, I did, too. And that didn't really change over the next several years, in spite of my neat, peachy-keen course reader.

My first attempt at a Current Reality Tree was on the topic, "Why Total Quality Fails in Organizations." It took me the better part of five days in New Haven to finish this tree. And the result was somewhat depressing: perhaps 150 entities and 40 Undesirable Effects (UDE) on small Post-it notes, stuck on three or four huge sheets of butcher block paper. But, damn it, I had a core problem! A single core problem that said, in essence: "It's the fault of bad leadership."

What a revelation—not! In truth, almost *all* problems in complex organizations can be laid at the feet of leadership. I didn't need a 150-entity logic tree to tell me that! Just as bad, I found the logic trees my grad school students had built were equally vague. Lots of entities, lots of Undesirable Effects, and over-broad core problems. As the years went on, I realized that the same was true of my clients.

What was lacking?

Focus

I'm glad you asked that question. The answer is focus. My Current Reality Trees, as well as those of my students and clients, were "all over the waterfront."

It shouldn't be this way, I thought. We drown in Undesirable Effects, many of which didn't really seem all that important. We're saddled with broad statements of core problems that took on many of the characteristics of ... marshmallows. To a "systems" guy, there seemed to be precious little ... well, system ... in it all.

I tried to think of a way to impose some focus on the whole process. In my Thinking Process courses for clients, I started them aiming for a well-defined problem by having them create a title for their Current Reality Tree that started with *why*. "Why does our organization [enter your own choice of negative outcome]?" From there it was easy enough to get the participants to create statements demonstrating that the negative outcome was happening. It was a start.

But even then, the guidance for determining Undesirable Effects was too broad. The Goldratt Institute answer to "How to come up with UDEs?" was to write down "anything you don't like that's happening in your system." Well ... *that* doesn't exclude very much— I can bitch and moan all day about things I don't like in *my* system. And others could do the same.

There was another problem with Current Reality Trees, too: *How do you know when it's time to stop "drilling down" in the cause-and-effect chain?* The "book solution" (from 442 Orange Street) was, "Stop when all your cause-and-effect branches have converged into a single entity— a *core problem*.

Sorry, but *that* wasn't much help, either. If I wanted to merge all the branches of reality in *any* situation, I knew the core problem for that. It sounds like, "And God said, 'Let there be light.' " If that isn't the single root cause of everything wrong with any organizational system in the world, then I don't know what is. But there's a problem with that statement of a core problem: it's not exactly *actionable!*

So, clearly, for any CRT on any subject to be actionable, we have to draw the "drill-down" line *somewhere*. And that "somewhere" is almost guaranteed to be *above* the level of a unitary core problem. But what does that leave me with? Multiple branches of the logic tree that *don't converge* into a single core problem.

This whole challenge of getting problem analyses—that is, Current Reality Trees—focused was becoming a real pain in the ass. There had to be a better way...

A Clean Sheet of Paper

The only thing I could think of to do was to start over, from scratch, with a system perspective. Fortunately, the Theory of Constraints provided me the "seed" for doing that in a book called *The Goal*. And what was that seed? The goal, of course! (Duuuhhh...) I began by asking, *What's the goal of the system?* Sorry, but it's not as simple as parroting "To make more money, now and in the future." There are a lot of systems out there, and many—perhaps most—of them are *not* in the business of making money.

Nevertheless, defining the goal of the system to be analyzed seemed a good way to start. Fine. But what comes next? It was at this point (around 2001) that I recalled something I'd seen in a short program at the Goldratt Institute in 1995. As I sat around the conference room table with a half-dozen others, I noticed something on an easel that was not part of the program. It looked like a very “busy” Prerequisite Tree, but it didn't have any obstacles. I asked about it and was told, "That's an Intermediate Objectives Map." I thought no more about it ... for six years.

Then, as I struggled with what to do next with the goal statement I'd started with, I remembered the Intermediate Objectives (IO) Map. *What happens if I put the goal at the top of an IO Map,"* I wondered? Boy, that was easy! But now... what? Think, Dettmer! *What's directly under the objective in a Prerequisite Tree?* Well, whatever it is, it's the *last thing* that must happen before the objective is attained. So, what's the last thing that must happen before the goal of a system is realized?

Critical Success Factors

Without warning, from deep in my memory banks, the title of a *Harvard Business Review* article from 1979 crawled into my consciousness. "Chief Executives Define Their Own Data Needs." [1] It was the very first mention of "critical success factors" that I ever saw. I dug it out and re-read it ... for the first time in 20 years. And to my dismay, though it used the term, it did not define what a critical success factor was! It offered something more like "I'll know it when I see it." Not much help there. But it *did* get me to thinking, since the article wasn't any help, how would *I* define a critical success factor?

One criterion seemed obvious—if it were to be truly critical to goal attainment, it would have to be something that happened *just before* the goal was realized. In other words, it had to be the summary outcome of all contributing activities in the system. A *terminal outcome*. And that terminal outcome would have to constitute an important component of the goal itself. Now, we're getting somewhere!

But ... how many critical success factors does it take to attain a goal? Clearly, it can't be a lot. The more things one designates as "critical," the more diluted criticality becomes. If you have ten things you consider critical, then *none* of them are truly critical.

Let's go back to the goal for a minute.

Goldratt suggested that the goal of a company (for-profit, we'll assume) was to "make more money, now and in the future." Okay ... in the final analysis, how does that happen? Well, Throughput Accounting gave us a pretty good answer to that: Maximize throughput, minimize inventory, and minimize operating expense. [2] *There you have it,* Dettmer. What are these but critical success factors that define a goal — profitability! They're most assuredly terminal outcomes of many subordinate activities. They're definitely high level, because they happen *just before* the goal is attained. And there are damned few of them!

But what if a system's goal is not to maximize profitability. Take, for example, a fairly autonomous sub-unit of a larger corporate entity that completes important work enabling other parts of the system to complete their work. Like, maybe, Henry Ford's Rouge River steel plant, which smelted the steel for all those Model-Ts that Ford sold between 1908 and 1927. Rouge River could have been said to have a goal and critical success factors, couldn't it?

Yes, it could. But that goal and critical success factors wouldn't necessarily have been financial, would they? At least not all of them. The goal would likely have been tons of steel, and in place of throughput (a financial term) there would have been some other non-pecuniary metric. Inventory and operating expense could still be relevant as fiscal metrics. We won't get into those details here. It's enough to say that even in a non-financial situation, the number of critical success factors would be limited. For me, three seemed a good starting point.

A Goal and Critical Success Factors — Now what?

Okay, now I've got a goal and three critical success factors. *What do I do with them,* I wondered? Here's this Intermediate Objectives Map sitting here, doing ... not much. What if I arrange the goal and critical success factors, in two layers, and plop them on the IO Map as the

top two layers? Then what are all the layers below? Clearly, like intermediate objectives in a Prerequisite Tree, these elements are necessary in order to realize the critical success factors. *Necessary conditions*. And there's a logically dependent network of them below the top two levels of this new type of tree.

So there we have it: A system goal, dependent on three (or perhaps more) critical success factors, which themselves are dependent on a substantial network of necessary conditions. I wondered, *What do I call this?* It's not a Prerequisite Tree, because its purpose is not to overcome obstacles. And it's a very high level tree. After all, it culminates with the system's goal. Since it was an adaptation of what I knew as an IO Map, and since it clearly overarched the system — however you define that — I referred to it as a "strategic intermediate objectives map" ... or S-IO Map.

And that's the way it first appeared in print, in my book *Strategic Navigation* (2002). It wasn't until eight years later, when I was constructing one of these trees with Deloitte Consulting, that Martin Burns, a Deloitte partner, began referring to it (to senior management) as a "goal tree." And once again, I had that "Duuuhhh..." moment. *Why didn't I think of that?* So, realizing which side of my bread the butter was on, it became a "goal tree" there ever after.

Emerging From the Rabbit Hole

Nice story, that, about how Goal Trees came to be. But I got woefully off-course. Before I went down that rabbit hole, I was trying to improve the quality, and to a lesser extent the speed of completion, of Current Reality Trees. My intention was to make them easier and faster for people to "get right" the first time. How does a Goal Tree help do that? Back on course, now...

If you recall, the disorganized nature of Current Reality Trees was an entirely predictable outcome of the over-broad definition of an Undesirable Effect. With the Goal Tree now in hand, it was time to compare it with reality. And the first, most obvious take-away from that was that a new definition of Undesirable Effect was essential. No more "anything you don't like about your system." Instead, to keep attention focused on the system (instead of petty aggravations), an Undesirable Effect would be a *failure to satisfy a critical success factor, or the goal*.

And since there were very few critical success factors (usually no more than three), there really couldn't be more than the same number of *system* Undesirable Effects.

Whoops! Add *one more*. According to the definition of a critical success factor, failure to achieve it automatically results in failure to achieve the goal. Since all the CSF are critical to success, a failure on *even one* means goal attainment is compromised. After all, that's the essence of necessary condition logic. So, if your system is failing at one critical success factor, there are really two Undesirable Effects — one for the CSF and one for the goal its failure-to-achieve prevents. A *really bad* system might fail at all of its critical success factors, in which case it would have that number of Undesirable Effects-plus-one.

Boy, did this simplify Current Reality Trees! Instead of 30 or 40 Undesirable Effects, there were usually no more than four, and possibly only two! How easy should *that* make Current Reality Trees?! Pretty easy, as it turns out. What used to take four to five days to complete could be accomplished in *four to five hours*. What was the difference? *Focus!* Instead of trying to connect 30 or 40 UDEs, people were only having to connect four or fewer. That's *bound* to take less time.

Not So Fast …

There are, however, a few new considerations in this method of defining Undesirable Effects. With the “old” way of constructing Current Reality Tree, the objective was to drill down

to a *core problem*. By definition (of 442 Orange Street), a core problem was one that accounted for 70% of the UDEs in a system. At first blush, that might sound okay—if you can't get 'em all, focus your efforts on the majority.

But there are a few problems with that. First is that it assumes that "all UDEs are created equal." That is, each of them are equally bad. When the criterion is "anything you don't like," that may be okay. But what if 70% of your core problem's UDEs are of the "my in-box is too full of paperwork" variety, and the failure to generate enough sales for the quarter ends up in the remaining (untouched) 30%—the ones you're ignoring because they don't result from your core problem?

The second consideration is that with the new definition of UDEs—a failure to achieve a Critical Success Factor—is that if one such UDE can prevent Goal attainment, then *any* of them can. This means you can't deal with only 70% (a core problem). You have to get rid of *all* Undesirable Effects. Fortunately, there won't be as many … no more than the total number of Critical Success Factors, plus one. That's a much more manageable number.

A third consideration is the motivation to search for a core problem—the one major deficiency that can be "shot with a silver bullet." As I intimated earlier, the search for a core problem might result in a very "fuzzy" problem statement. That's bad enough. But it also might lie beyond the authority of the chief decision-maker to do anything about. What do you do if you have three branches in your CRT, you're dying to merge them into a nice, neat core problem that you can make go away and solve all the system's ills, but you "ain't got no control"…?

The answer is, you're faced with working on several root causes. And to the extent that these causes ultimately produce UDEs 'way up above, they're critical to resolve. You might call them *critical root causes*. At least, that's what I call them.

So, if you're applying this later-generation approach to building CRTs, you have two requirements to keep in mind. First, you have to eliminate *all* UDEs, not just 70% of them. (Fortunately, there aren't as many as there were before.) Second, you must accept the fact that in all likelihood you're going to have to deal with *multiple critical root causes*. Maybe only two. Maybe three or four, but it's not likely to be more than that. Not if you only have three or four UDEs at the top.

But ... but —

I know what you're thinking: *Where did all that detail from the original CRT approach go? It can't have just disappeared!*

No, it can't. And much of it didn't. (Some of it did, because it wasn't necessary-and-sufficient for the new, more streamlined CRT.) Instead, if it was truly relevant to the streamlined causality, and much of it was, it found its way into the chain of cause-and-effect lower in the tree, not at the UDE level.

Lest you think that CRTs became superficial pictures of complicated reality, think again. They could very often still have 50 or 100 entities. But the branch structure was far more robust and less confusing. By way of example, here's the "silhouette" of a post-Goal Tree Current Reality Tree. (**Figure 1**) It describes the problems of a $10 billion financial corporation. Was it too confusing? Hardly. The CEO's reaction: "I've been looking for this for 25 years..."

The Bottom Line

That, as Paul Harvey would have said, is the rest of the story. Now you know how the construction of Current Reality Trees went from five days to five hours. And how I was able to

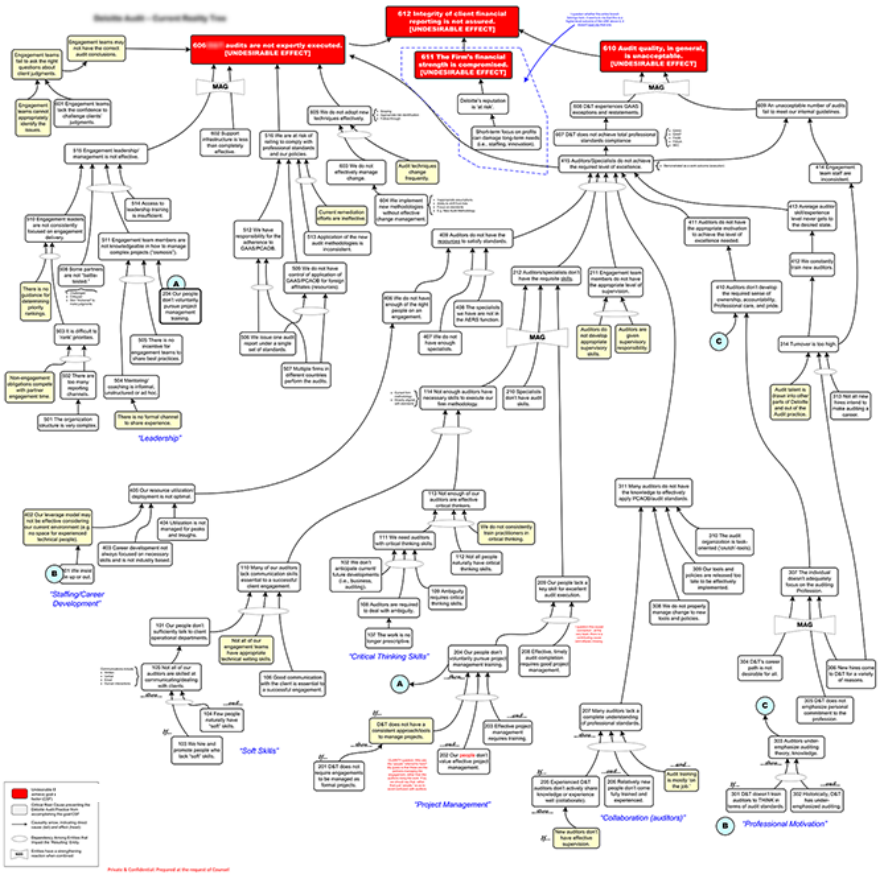

Figure 1. Current Reality Tree — Financial Services Company

compress a 10-day Jonah course into five days.

But along the way, the Thinking Processes "morphed" from the configuration and structure originally conceived by Goldratt into a newer, faster, easier-to-accomplish model — one that brings with it better system focus the first time and more satisfied customers in the final analysis. My approach was so radically different from anything that any other students of the Goldratt Institute Jonah Program were doing, I felt the need to distinguish it from the original. So, I refer to what I teach as the *Logical Thinking Process* (singular), instead of the Thinking Processes (plural).

Versatility Magnified

Moreover, I found a curious thing happening. Although I originally intended the Goal Tree to be merely the precursor to cleaner, more robust, focused Current Reality Trees, as time went on I discovered *other* benefits to it, too. I found that the Goal Tree could stand alone — without a complete thinking process analysis following it. It could serve as a constant "beacon" for organizations to maintain their day-to-day focus on what they ought to be doing to succeed in their chosen field. A good example is Aerosud SA, an aviation subcontractor in South Africa. Their managing director had their Goal Tree printed in large scale (**Figure 2**). It now greets visitors and employees alike in the lobby of their corporate offices — a constant reminder to both of their focus on what it takes to succeed.

I discovered, too, that the top two layers of the Goal Tree — the goal and the critical success factors —could be applied to the development of Evaporating Clouds for conflict resolu-

Figure 2. Johan Steyn, Managing Director of Aerosud SA, and Andre Tustin, Continuous Improvement and Supply Chain Synchronization Manager, and the company's Goal Tree

tion within systems, making their creation faster, easier, and more effective the first time. (**Figure 3**) And the same two layers of the Goal Tree provided part of the "skeleton" of a Future Reality Tree, making that tree faster, easier, and more effective in creation.

Summary

What do I want you to take away from this little history lesson?

- First, the Goal Tree is the core of the Logical Thinking Process. It informs *every other* logic tree.
- Second, the Goal Tree has many of the same characteristics of a Prerequisite Tree: it culminates in a single outcome (the Goal); it starts with a network of sequential necessary conditions; and it may have parallel branches that cross-connect.
- Third, it employs the concept of Critical Success Factors—a limited set of *terminal outcomes* (necessary conditions) that define goal attainment. In doing so, it provides the system -level focus absent in the original thinking processes.

It's worth noting that, contrary to what some people might think, the vertical placement of necessary conditions and Critical Success Factors in the Goal Tree implies *nothing* about their importance relative to one another. The nature of the necessity-based logic makes the lowest necessary condition in the tree equally important as the Critical Success Factor … because if you fail to accomplish a lower level necessary condition, none of what transpires above it, including a Critical Success Factor, will happen. That makes the key distinction between a Critical Success Factor and any other lower-level necessary condition the fact that the former is a terminal outcome of the preceding activities.

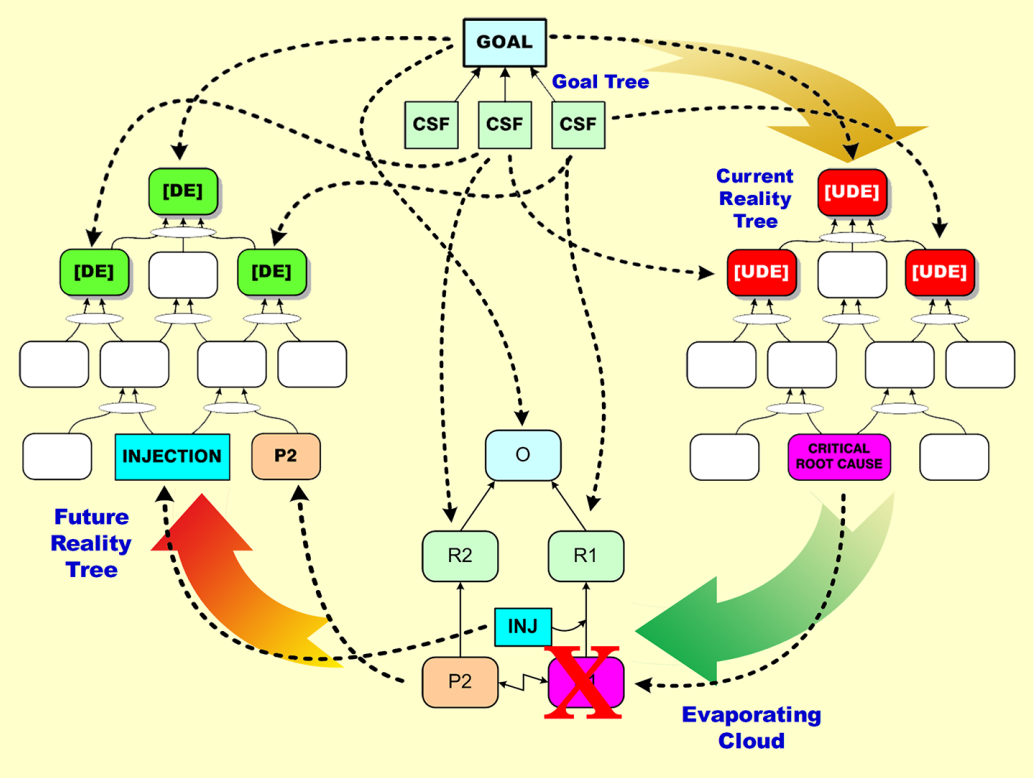

Figure 3. The Goal Tree informs not only the Current Reality Tree, but the Evaporating Cloud and the Future Reality Tree as well. It's the key to the entire Logical Thinking Process.

The Goal Tree really *is* the foundation for all the trees in the Logical Thinking Process. And after nearly 30 years, I can still say that the Logical Thinking Process is the best policy analysis tool ever created. And the lion's share of the credit for that goes to a physicist who didn't let the field of physics constrain his creativity — Eli Goldratt. ♦

ENDNOTES:

1. Rockart, J.F. "Chief Executives Define Their Own Data Needs." *Harvard Business Review*, March 1979.

2. These were Goldratt's original words, from *The Haystack Syndrome* (1990). Over time, I (and others) have concluded that "optimize" would be a more accurate characterization than "minimize."