

HOW A CEO CAN CREATE A SUSTAINABLE COMPETITIVE ADVANTAGE BY MODELING THE COMPANY AS A DYNAMIC SYSTEM

by
Bob Berry
CEO
DPS Telecom

for the
31st International Conference
System Dynamics Society

Cambridge, Massachusetts
July 24, 2013

Abstract

Modeling a company as a dynamic system is a powerful tool to guide business improvements. Models provide "insight into the physics" of nebulous bottlenecks that restrict growth. This has been demonstrated repeatedly at DPS Telecom, where models have been successfully used to grow the company from \$6 million to \$13 million with no increase in staffing. Insights into client churn, product quality, company culture, and profitability were all modeled, understood and improved. The final model, dealing with self-regulation within the Sales Department, resulted in a multi-million dollar decision to buy out one of the partners. This paper starts by explaining why the CEO is uniquely well positioned to lead modeling efforts, and it gives a quick outline describing how to get started. Next, four models are introduced that helped DPS more-than-double sales, double sales per employee, and more-than-quadruple operating profits. Three of the models talk about sales and revenue. The fourth shows how modeling can help in soft topics like staffing and culture. Finally, a simple learning and execution flowchart is given to allow a CEO to hit the ground running and begin using this technology.

Modeling your company as a dynamic system is the fastest way to discover and implement a sustainable competitive advantage. When done correctly, it is a powerful way to get "buy-in" across the entire management team of your business. The "discoveries" your team makes are about their company specifically, which makes a big impact on the learning. This "buy-in" and company-specific learning leads to successful change programs.

Remember, this is an amazing technology that requires you to expend effort. Your model will have to be debugged and calibrated. Data must be collected, analysis must be compiled, and then results must be interpreted. Finally, actions will have to be planned and executed.

For a modeling exercise to be successful, we need someone who is good at looking at things broadly from the "10,000-foot view." We need someone whose bias is seen as fair and stretches across the entire scope of the business. This person needs to be known as both results-oriented and more interested in finding a solution than a scapegoat. Finally, we need someone who can gain access to everyone on the management team and take action based on the results of the model.

If we view the requirements above as a job description, who are we talking about?
It's the CEO.

Over the last 10 years as CEO of DPS Telecom, I have used modeling to target bottlenecks in the company. The models have helped me locate the "real" problem and quickly get input and buy-in from all stakeholders, which then led to very successful change programs. By starting with small projects, confidence in the process improves. I'm a part-time practitioner of model-building, and the impact on DPS is huge! Forgetting the millions in increased sales and profitability, the biggest impact has been on the company culture and the near complete removal of "Drama". My job is so much easier in a well-funded company surrounded by great staff.

GETTING STARTED

After reading Peter Senge's *The Fifth Discipline* and Barry Richmond's *An Introduction to Systems Thinking*, it seemed like simulation would be a great way to improve business operations by experimenting on a model. I read several books on simulation. The books had applications to business process improvement, but several issues emerged as I tried to apply it to my business.

At first, I struggled because I thought exacting detail was required in my models. There were too many quotes, sales, conversion factors, and shaky quantities to figure out. It seemed like we did not have enough data collected to be useful. Not only that, but mapping "real world" processes into acceptable "simulatable" models was daunting.

Eventually, I would run out of time with no model to show for my efforts. But even a failed attempt resulted in a lot of learning. I would meet with my staff, solicit their input, and mix it with what I had learned. Using standard improvement project tools, we would be able to kick up performance a notch or two, which ultimately was the goal.

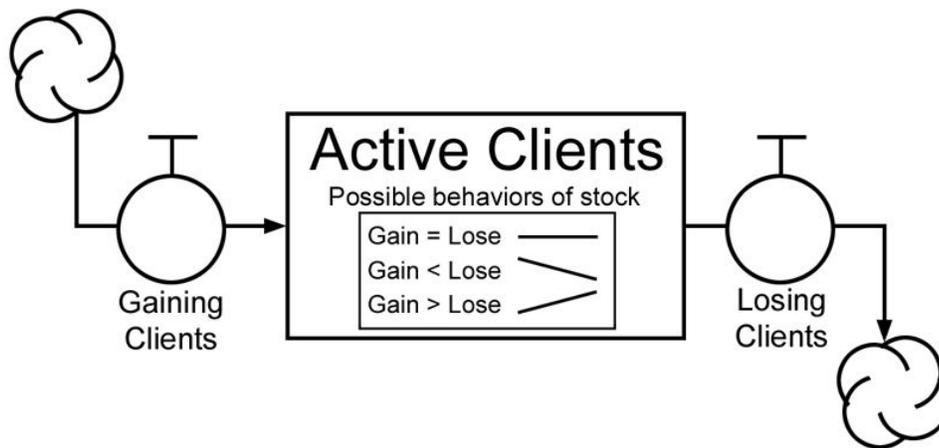
I kept looking for ways to use simulation, but I kept struggling with several issues. I didn't know how to map reality to the software. I couldn't figure out how to get acceptable metrics for sales, conversion factors, counts, flow times, and other key variables. Worst of all, I wasn't even sure what questions to ask of the simulation.

MODEL #1: CLIENT FLOW

My first success came during a management meeting. We were discussing the impact of quality issues on a client's repeat buying behavior. The debate got lost as it usually did. Some people said that we needed more first-time buyers. Others said that in-person site visits to our largest existing clients were required to stimulate repeat buying. The only thing most people were sure about was that more people needed to be hired. Our illogical collective opinion seemed to be, "I don't know what the issue is, but surely more expense and extra bodies will help!"

As I sat there, I could see we did not even have a grasp on the core issue before us. Some people were chasing symptoms. Some were chasing side effects. Others were chasing their own agendas. Everyone was angling to make sure it was someone else's fault. No one seemed to be interested in the cause.

After reading *An Introduction to Systems Thinking* several times, I was familiar with modeling concepts, even though I had not yet figured out how to use any of the software. So I drew a single stock with two flows:



We used this model to guide a discussion of our client base. Our quantity of active clients had been pretty flat for a couple of years. Using the simple model on the board, I taught the group about stocks and flows. It did not take long for them to get the concepts. We talked about the 3 possible states of the stock of clients (increasing, decreasing, steady), the results that generated those states, and what activities would lead to desirable results. At first, everyone was attracted to the "Gaining" flow. Some ideas were tossed around. But then they started playing with "Losing", and they quickly saw that growth would occur by closing the "Losing" flow - and that was a lot more under their control than the "Gaining" flow.

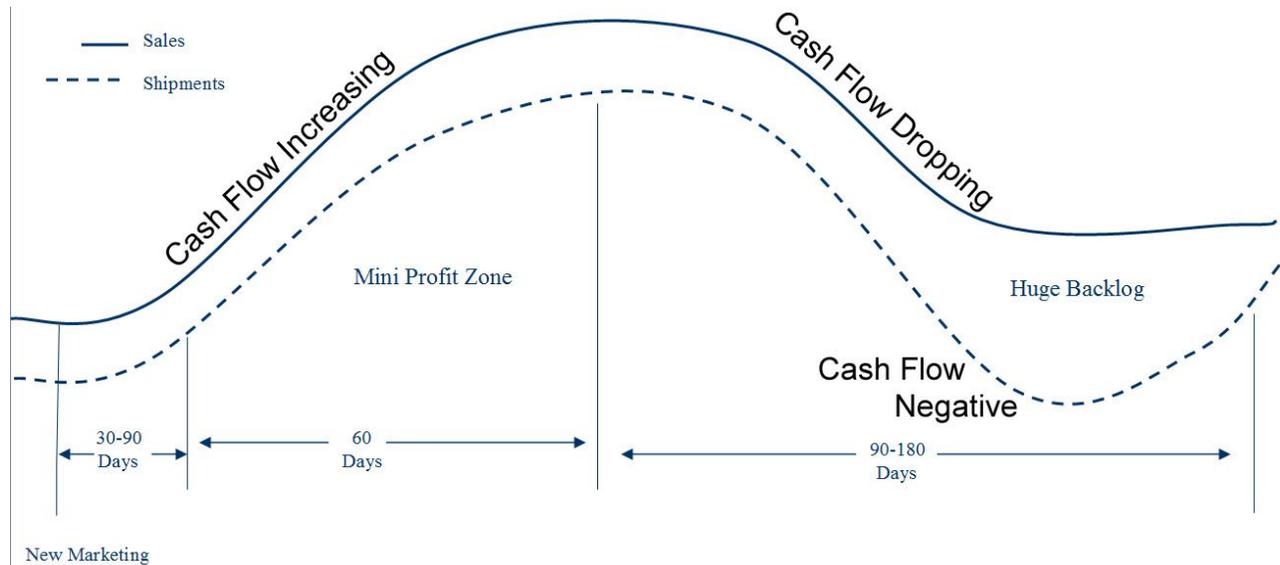
From that moment of learning, the entire team pulled together and created action plans. I got fast buy-in that changed our focus. That gave me the single largest group learning event that I had ever achieved with my management team. The takeaway? Team learning and the ability to drill to a core issue are key to moving a company forward.

This was a transformational event for me. I saw the power I had always dreamed about: The power to use computer simulation to help direct a group toward powerful change. And we had done it on a whiteboard! That single meeting changed the structure of my management meetings forever.

I was now more committed than ever to learning how to use this tool. The stock-flow concept was conceptually so easy to talk about, but abstract enough that we could talk about systems at a sufficiently high level.

MODEL #2: THE BLUE CRUSH

As the company started growing again, another recurring problem kept beating us up. It happened so often, we had a name for it: "The Blue Crush". We would introduce a new product, roll out a marketing campaign, and - if we were successful - The Blue Crush would nail us.



The Blue Crush is a big rise in sales followed by the inevitable crush of opportunity for Sales, Production, Engineering, and every other department. The company's processes are stressed by higher volumes, leading to delays and issues. Clients become upset. Sales spends more and more time talking to clients that have already bought and less and less time developing new business. This ultimately triggers a slowdown in new orders while we dig ourselves out of trouble - only to start all over again.

This was a good problem to finally force me into training for using the modeling software. About this time, Lexidyne was doing training for the isee iThink software. It was a 3-day event in Colorado with 2 days of beginner-level training, 1 day of intermediate training, and coaching available throughout.

I knew that something was wrong with my organization, but we didn't have a clear idea of "The Thing" that we needed to focus on. I had been in this situation before, and now I knew that I could clear away the confusion by building a model.

Before the class, we reviewed Marketing, Sales, Production, Engineering, and Tech Support. Again, I was looking for answers. I looked at sales figures, conversion factors, and other absolutes. I wanted "The Answer", but I had a sinking feeling that I wouldn't know what to do with the answer when I found it.

Inspired by the success and impact of my single stock model, I headed off to my first class. It's good that I was inspired because I was in for pain: brain pain.

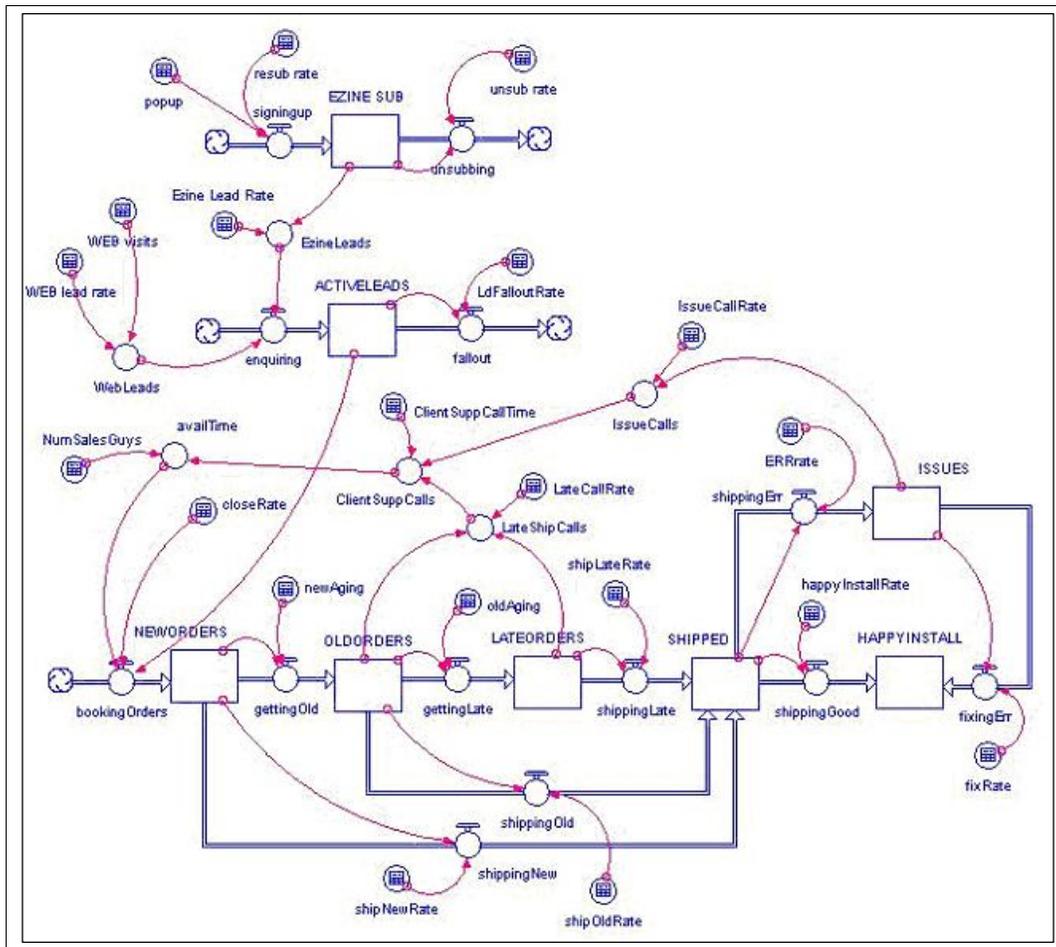
The class started by focusing on the key building blocks of system simulations. We went over the key building blocks (stocks, flows, and feedback loops) and how to use them to build main chains.

As I learned the fundamentals, I kept trying to map what I was learning to my problem area. Steve Peterson and Corey Peck were the class instructors. Whenever I would press Steve on how to get the metrics I needed, he would respond, "Use your best guess, or the best guess of the person in that area. You are not trying to simulate perfect reality. *You are trying to get insights into the physics of the problem.*" At the time, that concept was very frustrating.

It was a rough week. I kept trying to get something that was an accurate and detailed simulation of my company. Steve kept repeating that I should focus on the main effects: "You are trying to get *insights* into the *physics* of them problem." But I learned the tool, I learned how to debug, and I learned how to get multiple views of the model outputs.

After the 3-day class, I holed up for a week trying to build an abstract model of my problem. I still wasn't sure I knew how. I was really worried that, even if I got a model, I might not get a result that justified so much effort. But I thought that I might find a few useful "levers" to control things. I felt that I would only have control over hiring people and developing automation.

I learned a lot about simulation in trying to debug my new model. Quickly, I realized that, without the training on theory and the mechanics of the tool, I would have gotten nowhere. Finally, I had it running and, sure enough, there was the "Blue Crush" waveform. The wave manifested itself with my best guesses for values. It turns out that I didn't need exact figures; best guesses worked fine.



The Blue Crush Model

Now that I had this fairly abstract simulation that was producing similar results to my reality, I began looking around for the proverbial "So what?". What can I do with this?

My experience with debugging the model was invaluable here. Using the same techniques I used while debugging the model, I began peeking under the hood.

I kept pulling out different variables and graphing them until I found something interesting: As orders climbed, the sales team spent more and more of their time talking to people who had already bought. Instead of explaining our value proposition to new prospects and repeat clients, we were burning up our days by confirming orders, checking on late orders, and dealing with install issues and logistics. I suddenly had massive *insight* into the *physics* of the problem! Imagine that. Steve Peterson was right!



I could see the answer now, but would my team reach the same conclusion? I got the management team together. I spent several hours giving them some training on model concepts. Then we started playing with the model.

Of course, everyone wanted to plug in their favorite fix: hire more people. Just like in reality, no matter how many staff we added, the result was the same. It was "The Blue Crush."

So we looked and talked in more detail about the things that reduce Sales selling time. These include delays in Engineering, delays in Production, poor quality in Engineering, poor quality in Production, ease of our product installation, and the quality of our documentation.

Wow! We had quality issues. Only now we weren't dealing in abstractions. This was not some book talking about other companies. This was us. This was happening now, and we could see it was costing us money. We thought we had good processes in place already, but - with this model - we could tell we needed *BIG* improvements.

Empowered by new *insights* into the *physics* of our problem, we introduced LEAN process concepts into our Engineering and Production processes. This led directly to a 100% increase in sales over the next 12 months.

MODEL #3: MTM AND CULTURAL EFFECTS

As part of the LEAN implementation, we began documenting best practices for all key jobs. A new problem emerged: Drama! When people have gotten used to hiding laziness or fuzzy thinking within an undefined process, efforts to document that process trigger Drama of all kinds.

We had the full Drama spectrum: poor attendance, infighting, staff that did not live up to potential, excuses, missed deadlines, and (of course) staff turnover.

A pattern began to develop. At the center of each problem cluster was a person that seemed to have good potential but always seemed to have great excuses, generated lots of Drama, and seemed to hoard knowledge. They were heroes one minute and villains the next.

This pattern happened so often that we named it "MTM" - someone who "Manages Their Manager." The management team debated endlessly about how to inspire, develop, and motivate these people.

Just like with The Blue Crush, I was facing a pattern that occurred over and over. That sounded like something that could be modeled.

I went to another 3 days of training and spent the next week off-site focusing on my problem. I created the model, debugged it, and reviewed the output. I quickly learned something new, and I knew that the management team was not going to like it.

To soften the blow, I structured our meeting as a guided group learning project. The meeting was held during one long night. I started out by reminding everyone about models, the basic building blocks, the construction of chains, and how to read the graphs the system would output.

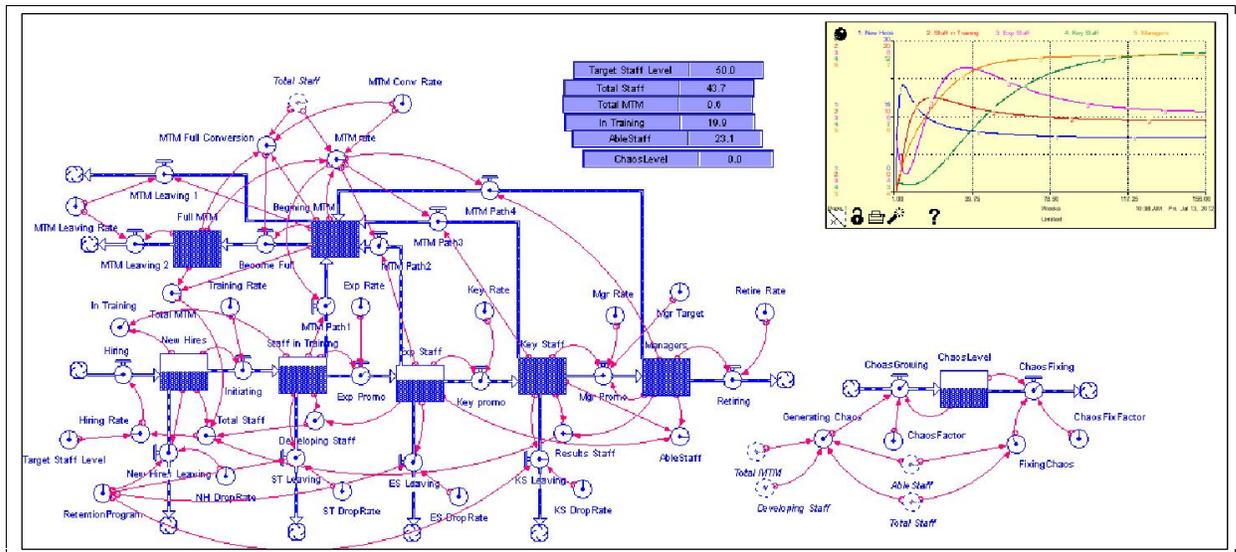
We then talked about our management, our results, and our staff issues. We talked about recruiting, training, and excuses. We discussed turnover and the role that Drama played. Finally, we talked about the various components of Drama and how it impacted performance of the Drama creator and everyone connected to that person.

Next, we brainstormed on "the mechanics" of our staff development and what the overall definition of success would be. I encouraged everyone to think five years ahead. Where were we, and where did we want to be?

Then we considered common MTM issues. There were employees who were clearly capable of doing the job, but always had seemingly good excuses for failure. Anyone who interacted with the MTM - even if they worked in a different department - was subject to "infection," triggering more Drama and turnover. MTM's hoarded the company's intellectual property and would not share with anyone. This made it impossible to scale up the MTM's area of the company.

Finally, we had enough information to build a model. I had already built one close to what we would need, so I built it again with the group watching.

We diagrammed the flow of staff development as a main chain, with a summary count of MTM's based on our observed MTM hire rate. We then treated the MTM count as an infection agent that fed into an accumulator we called "Chaos". We captured the mechanics of MTM's converting good employees to MTM status. We tied Chaos to the "Quit Rate" at each stage of staff development. Finally, we created a rate at which management fired MTM's in an effort to get Chaos and Drama under control.



The MTM Model

We ran the model over a 60-month period and looked at the results. With a little tweaking, we had a model that generated MTM results very similar to our current staff levels and population mix. It was depressing to see massive turnover, high levels of Chaos, and MTM staff outnumbering our well-trained staff 2-to-1.

After a couple of hours, we were all pretty disgusted with the whole issue. We had never seen with such clarity this waste of potential company accomplishments. Especially frustrating was the toll it had to be taking on people who were otherwise committed to the future of the company and the care of our clients.

We played with lots of different scenarios. We plugged in different "policies" and then ran the model for 60 months and looked at the effects on staff. In the end, we had several new conclusions:

- You cannot spot an MTM during their initial job interview.
- You cannot convert anyone once they become an MTM.
- A single MTM who trains your new employees can cause 3-5 of them to leave before their third year.
- The moment you discover an MTM, the only effective tactic is to "set them free" to follow their ambitions elsewhere.

With these policy settings, the model finally produced the world we wanted to live in someday. After 5 years, most of our turnover would occur in the first 6 months with about half again occurring in the second 6 months. After the first year, the turnover rate would slow very dramatically in Years 2-5+.

Over the next couple of weeks, we designed some real-world policies to move us in the direction indicated by the model. After a lot of brainstorming, some final conclusions bubbled to surface.

No Drama could be allowed anywhere in the company. We needed to help people exit quickly if they didn't fit the culture of the company. We had to eliminate hoarding of job

knowledge. We needed more cross-training, with all positions having at least 3 trained people who can do the job. We needed to retain most employees who had at least one year of experience. At the end of 5 years, we would have 25+ people who were fully trained, fully committed, and fully able to do an excellent job.

We implemented the policies we had tested in the model and, over the course of the next year, Drama plummeted as the team's capabilities soared.

Our first step was to identify our MTM's. There seemed to be 3 broad types. The first group had a complete inability to do the job, but they had amazing social skills devoted to covering up that fact.

A second group had amazing ability, but they never really believed that the job was good for them. They were always so full of promise, but they never quite lived up to it. They threatened to quit fairly often if pressured. You could get them help (employees), and the help never quite worked out either.

Finally, we come to "the amazing excuse crowd." Something almost believable was always happening to them. It was never their fault. Computers crashed. Hard drives failed. FedEx lost boxes. Family members died with disturbing frequency. I began to wonder whether there was any limit on how many cars could be stolen from one individual.

In the past, we felt compelled to sort the MTM's by type, then work with each differently. But with our new model-driven policy, it simply didn't matter. Any MTM must be promptly set free.

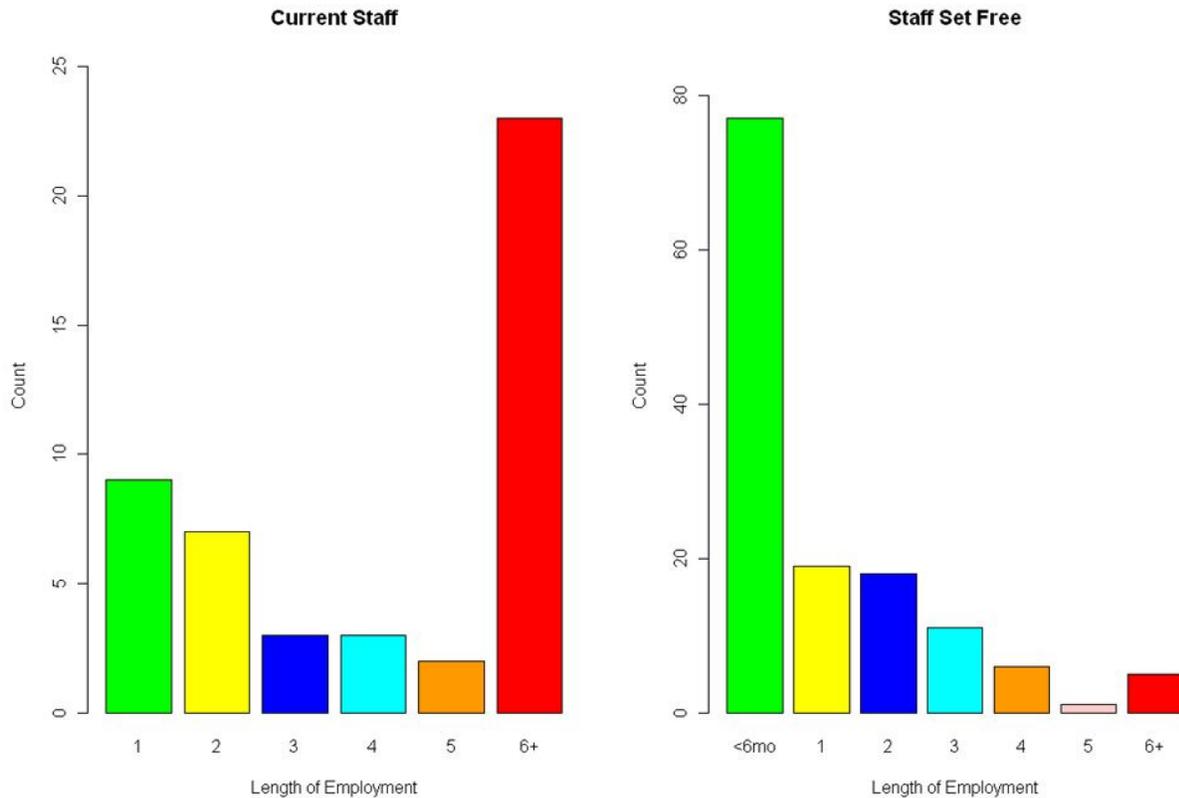
We started holding company-wide luncheons every month. At these meetings, the whole company hears news and plans directly from the management team, circumventing the MTM rumor mill. The luncheon also built a common vocabulary and exposed staff to upper-management goals and concepts.

We wrote standard operating procedures (SOP's) and conducted cross-training for every function in the company. This made scalability skyrocket, because the best way to make sure a position is scalable is by having someone else attempt the job by using only the documentation.

We tied quarterly bonuses to an individual's attendance, which fostered a culture of coming to work and focusing on the job. In fact, performance evaluation became very simple, because it was always tied to very specific results. We fully understood the nature of the job, the number of jobs the employee was cross-trained to perform, how well the employee contributed to SOP improvements, and attendance.

Finally, we really changed our recruiting process. Now, we were able to focus on existing skills and the ability to learn new skills. We added personality tests to enhance the interview discussion about culture, the expectations of the company, and the applicant's suitability for the job in question.

The best part was that, five years after introducing the model and the new policies it inspired, we were living in that place we once dreamed of. Most turnover happens in the first 12 months of employment, and half of our staff had at least 5 years of experience at the company.



MODEL #4: SELF-REGULATION

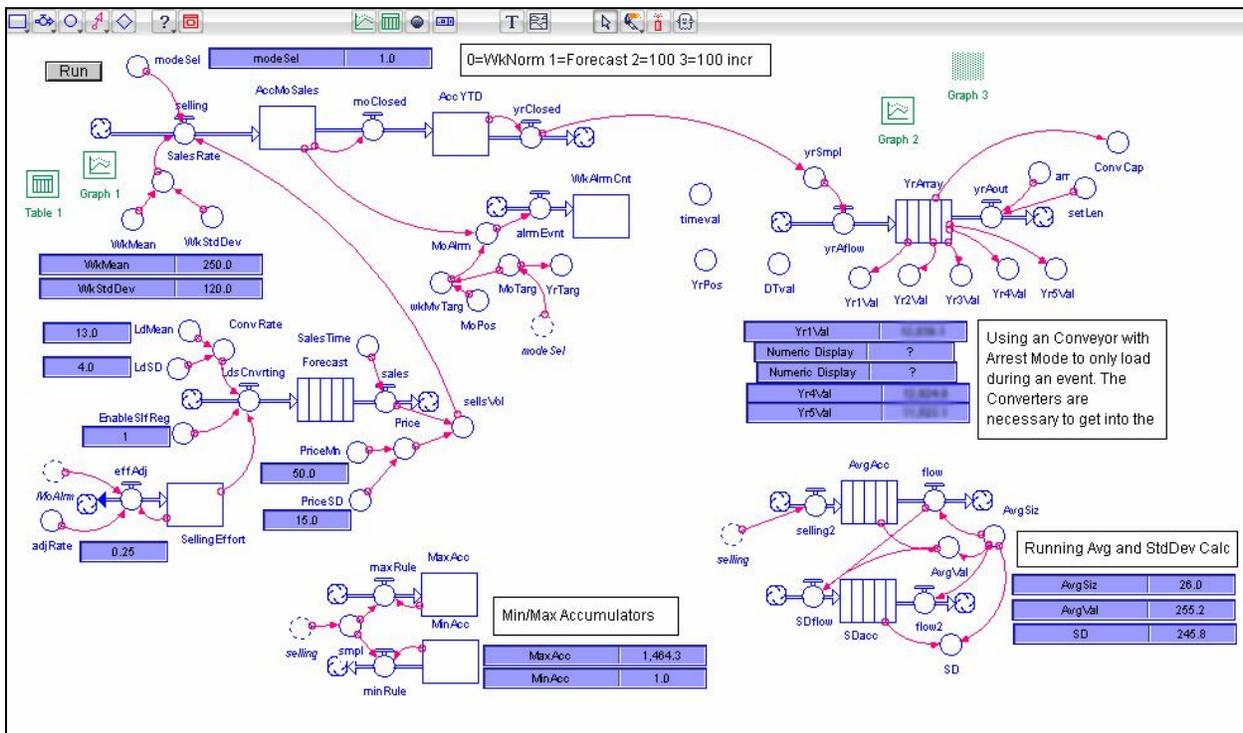
After a few more years, DPS was stuck again. Annual sales were stuck in the \$12-\$13M range. Our staff was good, our profits were amazing when considered as a percentage of revenue, but we were not growing.

Was this the result of Marketing activity, Sales effort, Engineering's product mix, Production's quality, or simply a lackluster economy?

Facing another tough problem, I headed off for another 3 days of retraining with Lexidyne and isee. I needed a refresher course on making models, including the building blocks, main chains, and the latest software updates.

I spent the next week in a hotel in San Francisco where I built a nice abstract model of the company. My hypothesis was that the Sales team and process were somehow a self-limiting, goal-seeking problem. I was curious whether I could build a model that was able to display the same characteristics and variability that I had observed in my sales metrics, quote volume, sales volume, repeat vs. new client ratio, and sales overtime hours.

After about 3 days, I had a model that emulated the company really well. Unsurprisingly, the only way to do it was to create a self-regulation accumulator that throttled effort and the resulting sales totals.



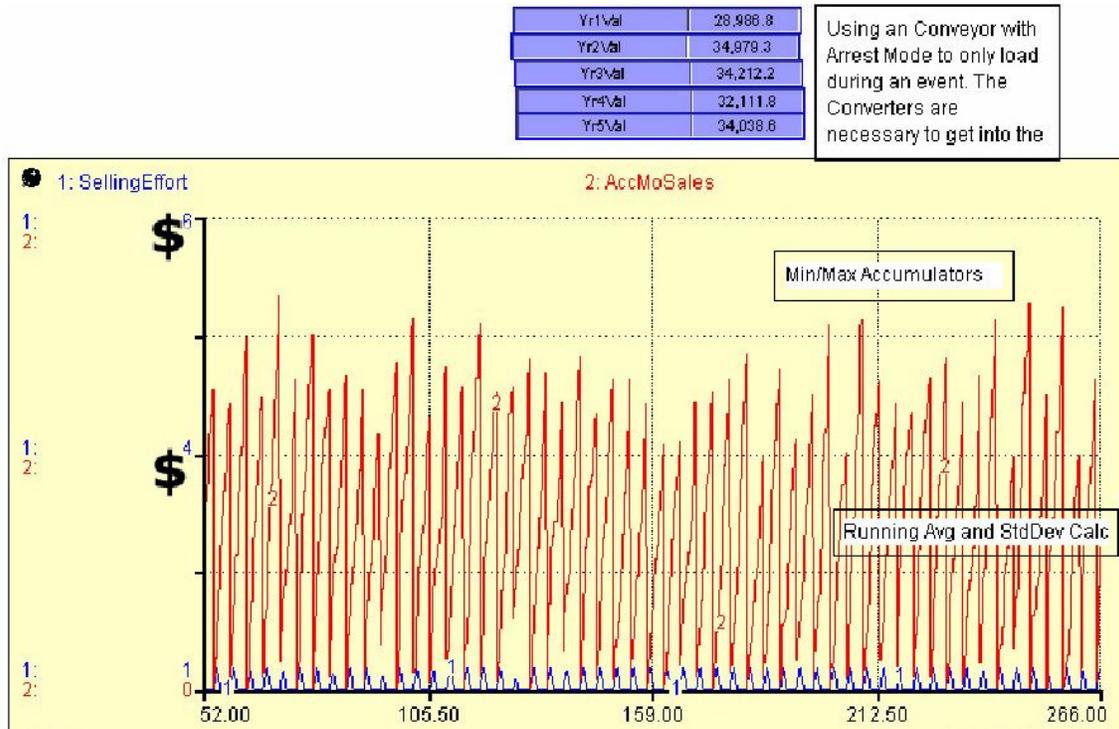
The Self-Regulation Model

Next, I ran numerous input scenarios to see if I could break the bottleneck. I added more and better marketing leads. I added new product mixes and new industry penetration. I improved the economic environment to be more favorable. Changing any of these inputs in my model had little to no effect on the output. That certainly felt like my reality.



In retrospect, the "economic environment" factor should have been a dead giveaway even before I started modeling. Regardless of what was happening in the economy, DPS always generated the same sales total. How could that be unless we were self-regulating our growth?

With the model complete, the first thing I did was turn self-regulation off. I wish it were that easy in real life, because our modeled sales instantly jumped to \$25-\$30 million per year. More importantly, it shifted the bottleneck away from the Sales Department. Suddenly, cranking up marketing, product development, or the economic environment would lead to an increase in sales results. That was my most important insight from my Self-Regulation Model.



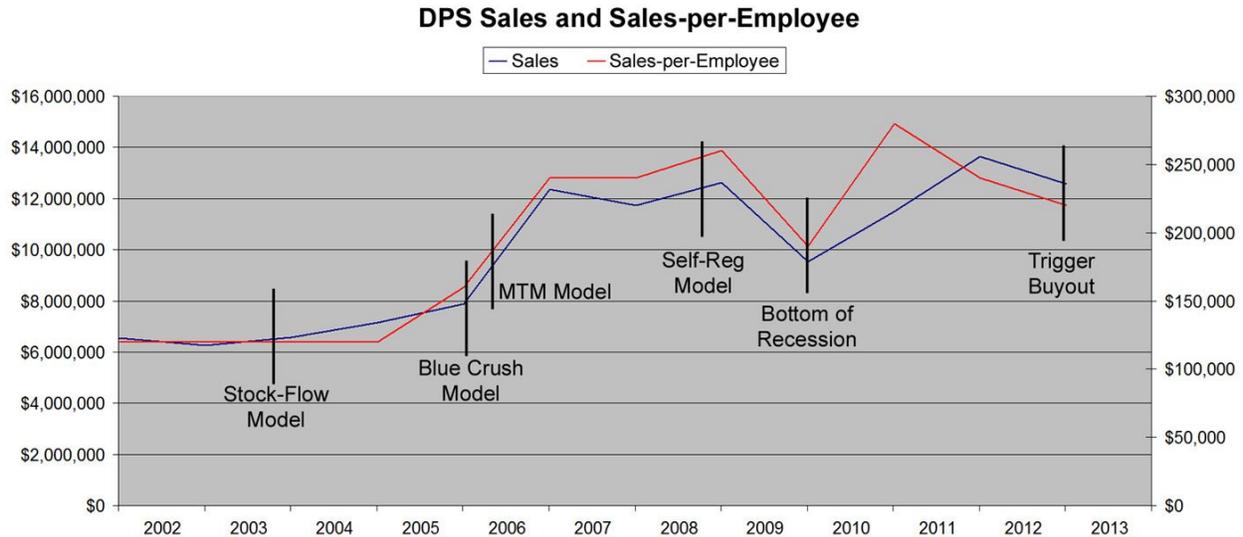
I returned to the office, met with the management team, and started changing the sales process. From the moment I had finished the model, I knew what I would have to do eventually. But first, I wanted to try a number of other things.

Over the next several years, I tried a lot of things. I changed the tools, the team, and lots of the processes, but the team could never scale. To their credit, the Sales management team tried very hard. Nonetheless, they could not grow our sales results. Finally, after once again finishing a year with the same sales total, I bought out my business partner who ran the Sales team.

My business partner is an amazing start-up guy. He's a design engineer who can sell, market, and negotiate incredibly well. He can do all those things so well that, even with his Superman cape tucked inside his business suit, he couldn't scale a team. He was simply too good himself. His ego and ability made it hard for others to develop under him.

So now, based on the results of another model, I have spent millions to buy him out in order to create the environment that will allow us to double the company in 18-24 months. That multi-million-dollar decision was easy for me to make because of the results of a model. I can honestly say that I believe in models.

THE FINANCIAL IMPACT OF THE MODELS



Modeling has been a big part of the success of DPS Telecom. The company had been stuck for a number of years around \$6M in sales. Every time I got the company growing, a number of things would go wrong. We are a high-mix, low-volume producer of remote-site monitoring gear, so customization is a big part of our product offering. As sales volumes would grow, the increase in workload would impact the quality and delivery times of products. This would lead to client frustration and would then impact growth negatively – leading to us being stuck.

For the **Stock-Flow Model**, I had called all the department heads together to talk about why we were stuck in this growth-quality cycle. The success of that simple model and the impact of the entire group learning together led changes that helped us break through the \$6 million barrier. Or, I should say, it led to changes that allowed us to move that barrier a little higher. By late 2005, sales had hit \$8 million but was stalling. Production was starting to have real capacity issues, and all departments were screaming for more staff. The Production backlog was at its highest level ever. The Blue Crush was coming, and it was bigger than ever.

In February 2006, after the isee training classes, we developed the **Blue Crush Model**. This model led the complete embrace of Lean Production throughout the company. A big part of Lean is the capture of Standard Operating Procedures (SOP's). This allows for faster training and cross-training of staff to cover whatever area of Production is swamped. As we started trying to document various jobs, we ran into serious HR issues – some staff simply refused to document their functions. This led to the creation of the **MTM Model** in April 2006.

With the conclusions we generated from the MTM Model, we chose to apply it to one department – the department in the most need: Production. I took 2 of the smartest people in the company, relieved them of their current responsibilities, and sent them to Production as “Change Agents”. We started in the most backed-up areas. People who would not participate in the SOP process were let go. Existing staff and new hires were trained via SOP, and - in a matter of months - the output of Production more-than-doubled, rework costs plummeted, and profits

more-than-doubled as well. You can see on the graph that Sales-per-Employee doubled as a result of the new Production capability.

All members of the management team were convinced of the viability of the MTM Model and the conclusions that we drew. Over the next year, the company applied it to every position. The impact on Engineering was dramatic. Both the cost of new product development and the time to market dropped dramatically.

Without the MTM Model, the LEAN process would never have been as successful. The result of these two models resulted in changing DPS from a 13% pre-tax profit to greater than a 45% pre-tax profit. Combine that with doubling sales, and you can see why I am so committed to modeling.

Finally, in 2008 I created the **Self-Regulation Model** to explain why we were again stuck in Sales. Despite a number of successful changes in Marketing and product offerings, we were again stuck. Profits, deliveries, and happy clients indicated that we should still be growing. But we weren't.

The model indicated that the Sales team was responsible for the limiting behavior – they had adopted processes that led us to being “locked in” at our current rate. We had no growth, but no decline either. I worked very hard with Sales to change their processes and staffing levels. But we could not get things to budge. I decided to wait until the end of 2009. If things did not change, I would have to make changes.

Then the Recession happened, at the end of which I remarried and had two beautiful babies. So it was not until the end of 2012 that I took action to replace my Sales Director. He was my business partner for 26 years and very successful, but DPS needed some change. The model said he had to go, but the 2 previous models made his valuation rather large! So he still is a big believer in models.

It's been a little over 4 months since the change, and our annualized run rate is currently calling for sales to break \$15 million for the 2013 year. The model says that we should be over \$25 million by 2014. We will see.

FINAL THOUGHTS

More CEO's need to learn about this powerful tool. I have been to 5 or 6 training classes for modeling, and I have never met a CEO or any other upper-management person in a class. I have talked about modeling to CEO groups. Most of them don't get it. Even as I talk about the impact on my company, only a few find it interesting. Of those few, none of them can resist delegating the work to an inexperienced person in Accounting!

Rudimentary skills can be learned by reading two books and playing with the software. The next step is going to a 3-day course and then spending 5 days alone to work on your current biggest recurring problem.

I cannot overstate the impact that modeling has had on my business and life. The culture of DPS is wonderful, as is our skyrocketing ability to create great products for our clients and great opportunities for our staff.

Culture has the biggest impact on your profits, quality has the biggest impact on your clients, and the bottlenecks that constrain your company will move whenever you grow. That's why senior management must drive the modeling exercise to keep it focused on useful conclusions. Then, they must follow through with action.

Modeling, however, is a tool that cannot be delegated. No other person in your organization has the right qualifications. In the end, the person who learns the most from this will be you. You will become a better CEO by building models.

All of your learning will be very specific to your world, and it will happen as you build and debug the model. You will explore and confirm the repercussions of the learning by playing out different scenarios. With the model, you can discover which areas are impacted by various inputs and policies. You will verify it in the real world, and you'll be able to design your improvement programs to impact those areas first.

Without the learning from the building and debugging, you can't hope to understand all of the mechanics. Without a solid understanding of the mechanics, you'll fall into the trap of looking for a "magic answer" from the model.

As you know by now, you're not looking for an answer from your model. "You're looking for *insight* into the *physics* of your problem."

Your team will also benefit by working with models. The group learning that goes into the modeling process is very inclusive. Blame and infighting tend to disappear as an "opportunity for improvement" mindset spreads within your team. Group learning is enhanced by the dispassionate and utterly fair referee that is "The Model". Everyone can be helpful by contributing their observations of the central behavior being modeled. This breaks down the barriers between departments and gets everyone involved who are typically isolated in their respective "silos". It also helps to prepare the group to learn from the modeling process. That is the key in order to craft the policy changes that will be implemented.

The process of modeling will lead you to a management team that is very analytical. They will begin to perceive the causes and effects within the big, slow-moving processes that make up your company. Your newfound analytical team will be able to periodically verify that your models still make sense as your worldview changes.

The ability to see, understand, and finally manipulate these slow-moving, ponderous machines creates a huge advantage for your management team. More importantly, you can achieve that most sexy of management outcomes: a sustainable competitive advantage.

A CEO MODELING PROCESS

Here is a straightforward modeling process you can use to get huge gains in your business:

- 1) Buy a copy of modeling software. I prefer continuous modeling.
- 2) Read a few good books on modeling business systems.
- 3) Look for small systems in your organization. Draw simple models. Share the concepts with your management team. Introduce them to stocks, flows, and converters. Plant the seeds that you will need later.
- 4) Look for a recurring problem that has been plaguing you. Draw a graphical depiction in a group brainstorming exercise with your management team.
- 5) Try to draw a model in your software. Struggle and expect "brain pain." At this stage, you are discovering things you "didn't know you didn't know". Review the books that you have read - see if you can get some clues.
- 6) Attend a multi-day training exercise.
- 7) Lock yourself away for a few days while your learning is fresh and work on your model. Get it working and get it debugged. Ask yourself, "What do I see?"
- 8) Talk to your entire management team. Walk them through your model, and see if they reach your same conclusions. Explore the model. Play with it.
- 9) Design an action plan. Implement your action plan and track your progress. Talk about the model a lot. Defer to the model to help when you struggle with new, unknown problems. Above all, remember to hold people accountable. Don't yield in your efforts to improve your world.
- 10) Annually, return to Step 4 and repeat!

REFERENCES

isee systems. Version 9.01 iThink: Systems Thinking for Business

isee systems. Training workshop: Introduction to Dynamic Modeling with STELLA and iThink

Lexidyne, LLC. Training workshop: iThink

McGarvey, Bernard. Hannon, Bruce. 2004 Dynamic Modeling for Business Management: An Introduction. New York, NY: Springer Publishing

Morecraft, John. 2007 Strategic Modeling and Business Dynamics: A Feedback System Approach. Hoboken, NJ: Wiley

Richmond, Barry. 2004 An Introduction to Systems Thinking. Lebanon, NH: isee systems, inc.

Senge, Peter. 1990 The Fifth Discipline. New York, NY: Doubleday Business

Sterman, John D. 2000 Business Dynamics. Columbus, OH: The McGraw-Hill Companies