

# Toolbox

In each issue, we present a different systems tool using relevant business examples. Readers are encouraged to practice using these tools by applying them to issues of personal interest.

## Fixes That Fail: Oiling the Squeaky Wheel-Again and Again...

In this issue, we continue our coverage of "systems archetypes"—dynamic structures which recur repeatedly in diverse settings.

How many times have you heard the saying "the squeaky wheel gets the oil?" Most people agree that whoever or whatever makes the most "noise" grabs our attention and will presumably be attended to first. The problem with following this adage is that it leads to operating in a reactive mode, continually "fighting fires" rather than making fundamental improvements.

To make matters worse, in our haste to grab the oil we often mistakenly pick up a can of water and splash it on the squeaky wheel. The squeaking stops momentarily, only to return more loudly as the air and water join

forces to rust the joint. We can stay very busy running about splashing water on all the squeaky wheels. But when there are finally no more squeaks we may discover that instead of having fixed all the problems, we have encased the wheels in rust.

### Better before Worse

In our search for the quick fix, we often rush into "solutions" without taking the time to understand the full impact of our actions. One of the tricky things about systems is that they usually point our attention toward short-term fixes and away from fundamental solutions.

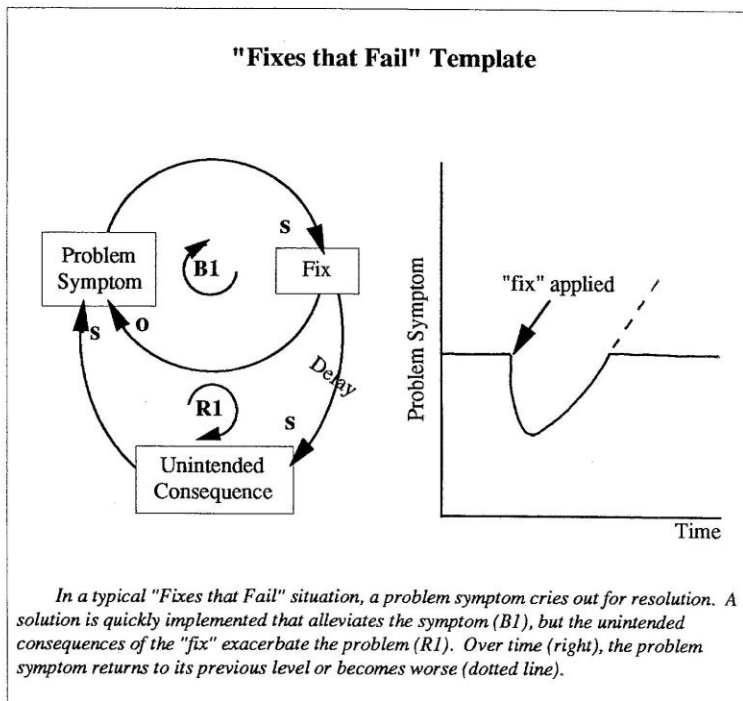
In a typical "Fixes that Fail" situation (see "Fixes that Fail" Template"), a problem symptom cries out (squeaks) for resolution. A solution is

quickly implemented that alleviates the symptom (loop B1). The relief is usually temporary, however, and the symptom returns, often worse than before. This happens because there are unintended consequences of the solution that unfold over a long period of time (loop R1) or as an accumulated consequence of repeatedly applying the solution.

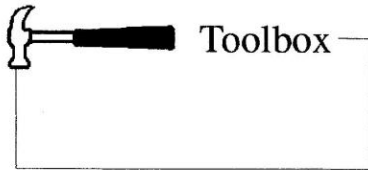
In the case of the squeaky wheel, the noise attracts our attention and we grab the nearest (easiest, most available, previously used, organizationally accepted, etc.) "fix" we can get our hands on and apply it. In the short term, even water will act as a lubricant and stop the squeaking. If we did not know anything about oxidation, we might assume that the water did, in fact, solve the squeaking problem. As the water evaporates and the metal oxidizes, however the wheel begins to squeak again. So we reach for the water again, since it worked the last time.

Of course we all know that oil or grease, not water, should be used to lubricate a squeaky wheel. But suppose the squeaky wheel is a customer screaming for a product that is two weeks late—how do we know whether we are applying the oil or the water when we respond? Do we understand enough about this situation's "chemical reaction" to take appropriate actions? Or, in our frenzy of fighting fires and oiling squeaky wheels, are we throwing oil on fires and applying water to the wheels?

**Expediting Customer Orders**  
Expediting customer orders, a common practice in many manufacturing firms, illustrates the "Fixes that Fail" archetype. A large semiconductor manufacturer, for example, is expediting



Continued on page 6



## Fixes That Fail: Oiling the Squeaky Wheel-Again and Again...

Continued from page 5

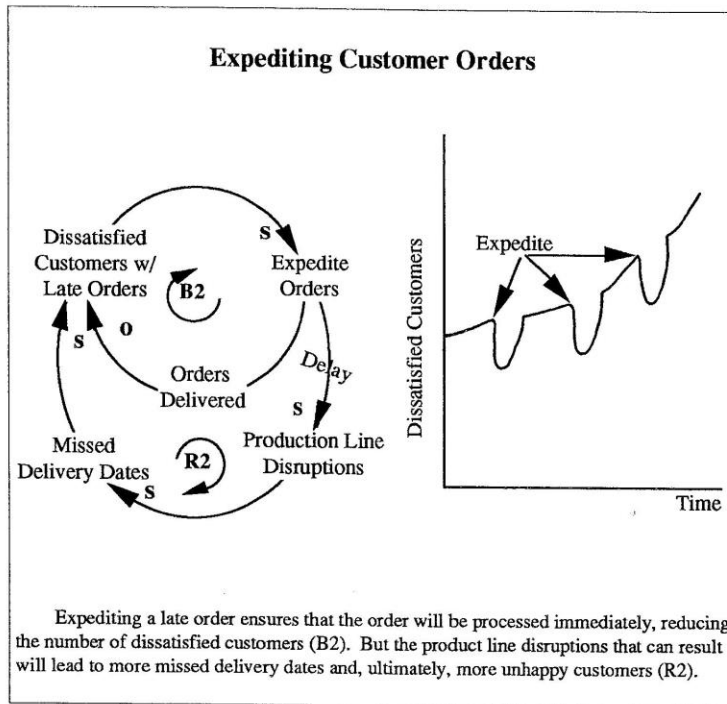
riencing some production problems and is running behind schedule on some shipments. They know that if their customers do not receive their orders on time, the customers literally will have to shut down their production lines until they receive the chips. So what happens?

Company A calls and demands that its chips be delivered immediately. The semiconductor company responds by assigning an expeditor to track down A's order and push it through the line ("Expediting Customer Orders" diagram). Of course it's not simply a matter of finding one item and escorting it to the loading docks. The company produces over a hundred different kinds of integrated circuits, and Company A has many different types on order. What's worse, the production steps from silicon wafers to final packaged circuits can number fifty or more. Finding and expediting A's order may mean wading through the entire factory and causing disruptions throughout the production line. Finally Company A's order is rushed through, resulting in a satisfied customer (loop B2).

But no sooner has A's order left the warehouse when company B calls demanding to receive *its* orders immediately, and the process begins all over again. At the same time, somebody *else* is expediting for company C. The squeaky wheels are getting oiled, but the number of squeaking wheels is rapidly increasing. As a result, the production line is continually being disrupted—leading to more missed delivery dates and more customer calls (loop R2).

### New Product Releases

Similarly, in a consumer electronics company, when one new product development project is in danger of



missing its release date, resources are often diverted from other projects to give it a final push. The product is released, but at a much higher cost. And as a result of all the shifting around, all the other neglected projects are more likely to need the same extra "push" in order to finish them on time.

Both in the semiconductor company and the electronics company, the quickest solution was to attend to each crisis as it happened. The specific problems were resolved, but at what expense?—a guarantee of more problems in the future.

### Using the Archetype

In most instances of "Fixes That Fail," people are usually aware of the negative consequences of applying a quick fix. But the pain of not doing something right away is often more real

and immediate than the delayed negative effects. If the long-term/short-term tradeoff was indeed one-for-one, where solving one problem today would create another one tomorrow, this strategy might be tolerable. But the reinforcing nature of unintended consequences ensures that tomorrow's problems will multiply faster than today's solutions.

Breaking the "Fixes that Fail" cycle usually requires two actions: acknowledging up front that the fix is merely alleviating a symptom, and making a commitment to solve the real problem now. Launching a two-pronged attack of applying the fix *and* planning out the fundamental solution will help ensure that you don't get caught in a perpetual cycle of solving yesterday's "solutions." ↻