



# Risk Reduction Through Inactivity

**Once again,** I find myself writing a BackTalk column sitting in an airplane. I live about two hours from an airport so between the joys of getting up early on a Saturday, traffic, airport parking, crowded check-in lines, totally full flight, and only a one-hour delay in taking off—I am just in a great mood.

Back in the day, this column was not called “BackTalk.” It was called “The Curmudgeons’ Corner.” A curmudgeon is defined as a killjoy, wet blanket, or a grouch. This column was a place to air gripes and general displeasure about software, bureaucracy, and the general process of producing software. There used to be

several authors—and all of us were grouchy. Over time, several of the authors decided to pursue other areas of literary achievement. You know what I think? They ran out of things to gripe about. Luckily, the publishers still have me around. I can always find something to complain about—such as change.

The story goes that a new assistant at a grocery store noticed that hardly anybody was buying rutabagas. It was the lowest-selling item in the produce department. He decided to remove the entire display of rutabagas. When the general manager noticed that the rutabagas were missing, he spoke to the assistant



and asked why. The assistant explained, and expected the general manager to compliment him. Instead, the general manager, with an irritated look, asked, “Well, why stop there? After all, now something else is the lowest-selling item!”

With that epiphany, the assistant realized that he could always remove the lowest-selling vegetable, until nothing was left but apples and unhappy customers. With this realization, rutabagas were restored, and the system was left the way it was. Leaving things the way they are does not imply you do not care—it might also mean that the current system works well enough.

Back in 1998, I became a Personal Software Process (PSP) instructor. I had to take the class (and pass!) before I could be certified to teach. Overall, I learned a lot, and my coding quality and speed really improved. Part of the process was submitting a Process Improvement Proposal (PIP) every time as I completed the exercises. The PIP was to force me to come up with an improvement on my personal process for developing software. And I rebelled. I submitted several, but eventually I reached the point where I was pretty happy with my own process, and did not really see a critical need to improve. Nevertheless, my instructor demanded that I submit a suggested improvement for each remaining program. As I remember, my last few improvement proposals consisted of things like, “Keep a pencil sharpener closer to my desk,” “Use higher-wattage bulbs during design,” and my all-time favorite, “Stock up on beverages, chips and popcorn prior to coding.” I eventually passed—and am still pretty pleased with my coding process.

I am not saying PSP did not help me—it did. What I am saying is that eventually I reached a point where things worked for me. Why change what works? When you modify a process, it involves risk. The risk of changing things is that somebody might

be less satisfied. Here are two similar “risk rules” I have learned over the years:

- 1.** You cannot make all the customers happy. In fact, you really probably cannot make most of them happy. What you can do is try not to make too many of them very unhappy. Sometimes, nobody is happy. But perhaps few are miserably unhappy.

- 2.** For customers, it is much easier to wait until a change occurs and then complain, rather than being proactive about what they need up front. There are lots of reasons for this—but the main issue is that customers do not really know what they want—but they are quick to realize what they do not want.

Sometimes the best thing to do is to change nothing! Those of us who work for the government or any large organization know the golden rule of change, “Change is often substituted for progress.” Need to look occupied? For heaven’s sake, change something. Cannot find something to change? Then it is probably time to reorganize. To the outsider, it looks like progress.

Do not wake a sleeping baby. Do not change a process that minimizes the number of unhappy customers. Sometimes the least damage you can do is not change a single thing. Before making changes, weigh the advantages of leaving things the way they are. It might make some customers unhappy, but it also might make less of them very unhappy than any other action. You are not being complacent—you are being passively proactive!

And I plan on being a curmudgeon for a long, long time.

David A. Cook  
 Stephen F. Austin State University  
[cookda@sfasu.edu](mailto:cookda@sfasu.edu)