

The Gatekeeper's Guide, or How to Kill a Tool

Eugene Farmer

Over the past few decades, new tools that facilitate the cost-effective production of high-quality software have slowly gained ground. The following guidelines show how you can push any tool to the side, perpetuating the manual methods that have always worked in the past.

1. Make it optional

Introduce the tool to the users with the directive, "Why don't you try this and see if you like it?" They will immediately focus on all the things they don't like and all the inconvenience the new tool will cause them. In no case should you ever suggest to users that you'll judge their performance based on how well they use the new tool.

If kept outside the workflow, the tool will wither and die.

2. Economize on training

That new tool is expensive! A good way to compensate is to minimize spending on training. Convince the vendor that all your people need is a half-day introduction because "they will pick it up quickly." Make sure to

- limit training to a group lecture in a small, hot room;
- insist on a minimum of 50 PowerPoint slides;
- avoid any workshops or hands-on exercises; and
- keep real content and examples away from the tool.

If the users become sufficiently frustrated during training, they'll develop a dislike for the product before they ever have to use it in a real

work situation.

For advanced players: If the vendor provides excellent training, make the tool mandatory and the training optional. If the vendor tends to do a mediocre job of training, make the training mandatory and the tool optional.

3. Don't use it in a real project

Use the new product only in pilot studies. Even better, never let it out of the "tools group." If the tools group people really like it, you're home free. People working on line projects scoff at anything the tools people like. Studies show that if you can keep a tool in the tools group for six months or more, it'll never take hold in real projects.

4. Never integrate it

So long as you keep the tool "stand-alone," it won't be a threat. If kept outside the normal workflow, the tool will eventually wither and die. The tool becomes invisible because organizations only monitor and measure things belonging to the production process. If kept outside of metrics collection, the tool can never demonstrate its value. Eventually a bean counter will conclude that there's no ROI, and the tool will hit the chopping block during the next round of budget cuts.

5. Use it sporadically

Particularly avoid using the tool on a regular basis. Intermittent use coupled with minimal training will leave users clueless the second time they need to use the tool. If you keep the interval between uses at months or longer, each use will seem as difficult as the first, and sooner or later, people will begin labeling it "hard to use."

6. Make it part of a quality initiative

Everyone pays lip service to quality. In most organizations, users know that deadlines come first, and quality isn't second but about sixth. Once you label a tool as something that improves quality, users will quickly decode the message: when we get to crunch time, we can forget about this baby.

7. Marginalize the champion

A tool enters an organization because it has a champion. Do whatever you can to make this person seem like a misfit. Usually that won't be too hard because agents of change stick out. If you can portray the champion as some kind of ineffective, misguided intellectual, you'll have notched a really big win. In some cases, you'll not only kill the tool but also discourage the champion enough to leave, which scores double.

An important corollary is to ensure that no one influential gets interested in the tool. Senior managers sometimes think they have a vision of how the tool could transform the organization. This can be dangerous because it's hard to marginalize them. Here, the best approach is to filter out all such notions before they can take hold. Make sure senior managers never get to see the tool.

8. Capitalize on early missteps

Begin by setting the expectation that the tool will fix—overnight—problems that took years to create. Early in the adoption of any tool, snafus, and glitches will occur; don't let these slip by as normal occurrences! Make sure everyone understands that the new tool is costing valuable time and money. Because benefits aren't demonstrated immediately—you have to first finish something and measure the “before and after”—time is on your side. Make as much noise as possible. By highlighting and magnifying every problem, you'll give the champion even less time to make the tool successful.

9. Make a small investment

This time-honored technique—the “let's go slow here” approach—is easy to do in large organizations. If you can

keep the initial purchase commitment small, you'll have little sunk cost to write off when your tool-killing campaign succeeds. This key step lets managers cut their losses at the first opportunity without losing face. Never make an investment that would cause embarrassment if the initiative fails, and make sure that no one high up in the organization has any “skin in the game.”

A potent variation is to get evaluation copies for free. This lets you reject the tool at an absolutely minimal cost to your organization. Never forget that making a small investment sends a powerful message to the company about your commitment to the tool.

10. Exploit fear, uncertainty, doubt, laziness, and inertia

Always remember that people resist change. Their fears are your friend. Intimate that sooner or later the tool will—not might—replace them. Repeatedly stress that they're going to have to learn

how to do their jobs differently. If you can get them to hate the tool, you're 90 percent there. Because once you can demonstrate that “the dogs won't eat the dog food,” who can dispute that the tool has failed?

Large organizations have no reason to continue squandering fortunes on tools. If the old ways were good enough for our grandfathers, they're good enough for us.

However, every company has a few misguided souls who'll try to improve things by introducing tools to help people do their jobs “better and more cost effectively.” It's up to us, using these guidelines, to thwart them at every turn. ☞

Eugene Farmer works for a software tools company and is writing under a pseudonym. He can receive your comments via *IEEE Software* at software@computer.org.

