

# Change Your Organization (For Peons)

Jim Little  
Titanium I.T. LLC  
jlittle@wgz.com

## Abstract

*In this experience report, a programmer at a medium-sized software company explains his tactics for bottom-up organizational change. The tactics are accompanied by personal recollections of related experiences. Tactics are presented in three primary sections: the difficulty of making change; getting attention; and making a case for change. A brief section on the consequences of change follows.*

## 1. Introduction

During the dot-com bust, I was hired by a medium-sized software company. I'll call it Aardvark, but that isn't the real name of the company. Aardvark had an enterprise-grade software product and a professional services arm that would provide custom software development around their product. I worked for the professional services group.

For several years, I had been working as a process coach: someone who came in and introduced teams to object-oriented programming and the Extreme Programming methodology. At Aardvark, though, I was hired as a peon—a “heads-down” programmer who was expected to deliver within existing structures.

I came into Aardvark with a strong bias towards agile methods, particularly Extreme Programming. I had several successful projects under my belt using Extreme Programming. Furthermore, I *enjoyed* working on Extreme Programming teams. I found the camaraderie and sense of doing good work very rewarding.

At Aardvark, the culture was different. Matrix management and waterfall processes were the order of the day. The culture was one of control rather than trust. I chafed at this culture and wanted to change it. I decided to take Martin Fowler's advice to “Change your organization, or change your organization” [1] to heart. I would try to change my organization from below, and if that failed, I would find a different organization to work for.

Along the way, I kept a detailed diary [2] of my efforts. Over time, I distilled specific tactics for organizational change [3] from my experiences. In this report, I'll be summarizing my experiences and tactics.

An important note: this is an experience report, not impartial research. The tactics I list here are the ones that I think were valuable for me, but they're fundamentally one person's reflections on his own actions, with all the attendant flaws and errors of observation. I hope this is helpful, but take my words with a grain of salt.

After this paper was completed, I became aware of Mary Lynn Manns and Linda Rising's book on patterns for organizational change [4]. Many of the tactics in this paper are described in more detail in their book.

## 2. Change Is Hard

“Somebody said (and I think it's on Wiki somewhere), ‘People hate change.’ The first reaction is, ‘Yeah, well, *everyone* knows that!’ To which the proper reply is, ‘No, you don't understand, people really **really** hate change.’

—Anonymous [5]

### 2.1. Question Your Motives

I wanted to change my organization because I saw that they were having trouble delivering software on time or of good quality. They had a culture of firefighting that led to mistreatment of their programmers: unplanned holiday overtime, unexpected cross-country travel, and general lack of respect.

At least, that's what I told myself. Deep down, though, I wanted change simply because the company didn't work the same way I was used to working. I came in, saw that they weren't using Extreme Programming or another agile methodology, and was already half convinced that they needed change. Luckily, the other trouble signs were there. I'd like to think that if they weren't, I wouldn't have tried to institute change. Still, I should have thought more carefully about what I was getting myself into—and more importantly, what I was getting my client into.

That brings me to my first and most important tactic for organizational change. I'll call it tactic #0 to emphasize its importance. I didn't recognize this one until well after I had succeeded at changing my organization. By that time, I had the responsibility for the success of a very large and very risky project, with company fortunes potentially riding on my decisions.

**Change Tactic #0: Question your motives.** Why do you want to change your organization? Are you just fond of a particular way of working? Think about what will happen if your changes succeed, but only partially, and you're responsible. Is it worth it? Or would you be better off changing your organization by finding a better job?

## 2.2. Establish an Escape Route

It turned out that Aadvark was a good place to try to institute change. It had its share of problems, just like any company, but management was generally open-minded and easy to talk to. Even so, attempting to change the organization was the hardest job I'd yet faced.

From the beginning, I treated the job as an experiment. I convinced myself that I was an easy hire, even in a tough job market. Knowing that I could get out any time gave me the strength to keep on going. Perhaps more importantly, it gave me the courage to try: to say risky things, to push hard, and to not give up. I felt that I was risking my job by doing the things I was doing, but at least I had a plan if things didn't work out.

**Change Tactic #1: Establish an escape route.** Determine how you can successfully get out, whether that means finding another job, going back to your old duties, or something else. Think about how and why you would make your escape.

## 2.3. Have a Solid Support Network

I didn't end up needing that escape route. Actually, I was successful. When I arrived, Aardvark was heavily focused on a waterfall-based control-oriented methodology and an object-oriented framework as a solution to their problems. Four months later, they started their first Extreme Programming (XP) [6] project with me in the lead. Still later, the company started deemphasizing matrix management and control-oriented approaches in favor of individual responsibility.

I can't take credit for all of it. Actually, I'm not sure how much I can take credit for. My approach to change involved a lot of talking, a little bit of leading by doing, and a bit of essay-writing. I think the net effect of that was to change the way some people thought about software development, but I can't really be sure. Bottom-up change is truly thankless: it's unlikely that anyone will even realize what you did. As a peon, you don't have the ability to make changes: all you can do is hope that you

can convince *other* people to change. It's not easy, and you'll never know how much was your doing and how much was just coincidence.

I worked constantly at introducing change and went home frustrated nearly every day. I could easily see mistakes being made, but the power to make corrections was out of my grasp. I started to burn out. I started looking for another job after three months, and at the end of four, I was ready to move on.

One thing that helped me a lot was my online diary [2]. I got some positive responses early on that made me feel like I was doing the right thing. I focused heavily on my personal life, too, trying not to think about work too much when I was away from it. I found that I was more effective when I wasn't feeling stressed. I was less tetchy, calmer, and more articulate.

**Change Tactic #2: Have a solid support network.** Bottom-up change is frustrating and thankless. Be sure that your happiness isn't dependent on job satisfaction. Rely on family and friends, go home on time, and don't dwell on work problems away from work.

## 2.4. Find Small Pleasures

I also found that I could do things at work that made me happier. Rather than focus on everything that I couldn't fix, as was my inclination, I focused on what I could fix. I didn't discover this for quite a while, but I found that even a bad day would be better if I accomplished some small programming goal. My best memory is of a day that I successfully wrote a unit test and refactored a user interface component.

**Change Tactic #3: Find small pleasures.** Organizational change is largely outside of your control. Find small things at work that you can do every day that *are* in your control and that give you a feeling of satisfaction.

## 2.5. Don't Give Up

Overall, it seemed that my best efforts didn't have any effect. I was able to make small changes within my sphere of influence, but I had to actively push every little change, and even then, they only lasted as long as I kept up the pressure. I could talk about problems until I was blue in the face, but the only reaction I would get was a patronizing "Yes, I can see how that would work for *other* people, but things are different here."

And then... things changed. I was on my way out the door, job offer in hand. I was about to give notice when I was offered leadership of a large project. I told them that I would run it as an XP project and listed all of the controversial practices I would perform. Pair programming, test-driven development, refactoring... all

of the things that had previously been turned down as too radical. And they agreed.

**Change Tactic #4: Don't give up.** I made very little difference to the way things were done in the beginning, but my efforts slowly changed the way people thought about problems. This change was largely invisible... but one day, it passed some threshold, and I had multilateral support for a big change.

### 3. How to Get Them to Listen to You

#### 3.1. Respect is Your Currency

My first job at Aardvark was to replace a programmer on a project that was going through the testing phase of the waterfall process. My role was to make changes and fixes as they were identified so they could be delivered in a twice-weekly "hot fix." It was a demoralizing job. I was a peon, and knew it.

As a peon, I had no control over others. I could influence other people, but I had no real authority. This was a precarious position from which to introduce change. All I could do was to try to convince *others* to change. That led me to an important realization:

**Change Tactic #5: Respect is your currency.** The more people respect you, the more credibility you have. The more credibility you have, the more opportunities you'll have to change their opinions. Earn respect by your actions.

#### 3.2. Be Respectful

In order to earn respect, though, you have to be respectful. The next tactic is one of the most important in this paper.

**Change Tactic #6: Be respectful.** Don't ever look down at anybody, no matter what, not even in the privacy of your thoughts. People can tell. A differing opinion does not mean ignorance, and ignorance does not mean stupidity.

#### 3.3. Do Your Work

The main thing I did to earn respect was to be very good at my peon job. I fixed more bugs faster than anyone else on the team. I did it without gloating or making a big fuss. I just did the work and checked it in. I think this established my credibility with the other programmers on the team and with its project manager.

People are often impressed by my typing skills. I'm a fast typist and I use keyboard shortcuts extensively. After seeing me program for a little while, people will often comment on how quickly I type. I can see from

their reactions that it's a short step from "he's a fast typist" to "he's a really good programmer" in their minds.

I'm not trying to say that you should go out and practice your keyboard shortcuts. (But it's not a bad idea.) My point is that even the smallest things can change the way people think about you.

**Change Tactic #7: Do your work.** You weren't hired to change the organization. Do—no, *excel at*—the work you were hired to do. You'll impress people with your dedication to the job and you'll earn respect.

#### 3.4. Stay Within Your Sphere of Influence

My early efforts at change at Aardvark consisted of fixing simple communication problems. The very first thing I did was to physically pick up my computer and move closer to the other programmers. A few weeks later, I tried to address drawn-out daily status meetings by writing my status on index cards and posting them publicly.

Both of these early changes targeted the issue of communication, but on a small scale. I had identified communication as a serious issue at Aardvark, but I didn't try to address the underlying causes. Instead, I focused on things that were within my reach. In the first case, the only thing I had influence over was myself. In the second case, I had a bit more influence, so I tried to change the way my four-person team reported status.

**Change Tactic #8: Stay within your sphere of influence.** Don't try to change parts of the organization you don't have continual contact with.

#### 3.5. Cultivate Champions

As people's respect for me grew, I started having regular discussion with senior developers in the organization. Over time, that led to good working relationships with managers, and then senior managers. As my sphere of influence grew, the changes I could make grew as well.

**Change Tactic #9: Cultivate champions.** Find at least one person who works with you, has a larger sphere of influence than you do, and is sympathetic to your basic philosophy. Establish a positive working relationship with them and use them to help propagate your ideas. Be honest about it.

#### 3.6. Don't Criticize Everything

One problem I had was that I tended to get frustrated. I saw all the problems that I could fix "if they'd only give me a chance to fix them!" As a result, I would occasionally rant. I'd complain too much about all the problems I saw. This did nothing to help me, and it

permanently colored the way some people thought about me.

**Change Tactic #10: Don't criticize everything.** If you find fault with everything, people won't take you as seriously. Pick something specific to work on.

### 3.7. Repeat Yourself

My early efforts at change didn't take root. I wrote bugs down on cards and taped them to a whiteboard. They were largely ignored, so I took them down, but I took opportunities to talk the amount of time we spent on status meetings, when I could do it without sounding like I was whining. I kept using the cards, and whenever someone asked me for status, I pointed to the cards.

A few weeks later, I went on a short vacation. When I came back, the visible status was back! People were writing their open defects up on the whiteboard, just as I had done with my cards. A few days later, noticing that the whiteboard was getting messy, I converted it back to index cards. This time, the change stuck, and we continued using cards to the end of the project.

**Change Tactic #11: Repeat yourself.** It's going to take a long, long time for your suggestions and changes to sink in. Say and do things over and over in different ways to different people. Try not to be too annoying about it.

## 4. They're Listening... Now What?

Q: How many psychologists does it take to change a light bulb?

A: Only one... but the light bulb has to *want* to change.

### 4.1. Find the Gap

I don't have any specific strategies for introducing Extreme Programming or another agile process into your organization. Every project and organization is different.

I have one idea, though, that's a winner. I got it from a book on sales. Since I was essentially trying to sell people on new ideas, it's not surprising that a book about selling would be helpful.

In this book, the authors said that there are basically four mindsets that you'll encounter in corporate sales:

- Flat: The organization is fine and doesn't need change.
- Growth: The organization is fine, but there's opportunity for improvement.
- Decline: The organization is declining but can be brought back to level.
- Denial: The organization is declining, but the person you're talking to doesn't realize it. [7]

The point made in the book is that you can only change an organization that is growing or declining. You can't change an organization that is flat or in denial. Essentially, you can't *cause* change unless people already *want* change.

I found this to be true at Aardvark as well. I spent a huge amount of effort on change in the first four months and met with limited success. My best success was the introduction of cards to track defects rather than long status meetings. That was a big hit with programmers because it solved a problem for them. They were in "growth" mode: the meetings were fine, but they wanted to spend less time in meetings and more time programming. My card idea gave them the change they already wanted.

On the other hand, the cards weren't a big hit with the project manager. For him, the meetings were "flat:" they reported status exactly as he needed. No improvement necessary. He went along with the cards, but grudgingly, and only because the programmers pushed it.

My biggest success at Aardvark was being given leadership of a full XP-style team. Again, the company was in a mode that enabled change. Using the credibility I had garnered in the previous months, I showed them that their projects for a big client were in trouble. This put them in a "decline" mindset. Then they, on their own, looked at a paper I had written about XP, which talked about the same problems and how to fix them, and asked me to lead the project. I'm convinced that being in a "decline" mindset is what led to them asking me to lead the projects.

Was I manipulating people by putting them in a "decline" mindset? I wasn't trying to. The projects really were in trouble. I wasn't trying to manipulate anybody, but I was trying to tell the truth in a way that could be heard. If I had been manipulating people, I think it would have eventually backfired.

**Change Tactic #12: Find the gap between desire and perceived reality.** People have to *want* to change, and they'll only want to change if they think it will get them something they can't get otherwise.

### 4.2. Lead by Example

I was also able to cause change just in the way I worked. One thing I've noticed is that when I have a lot of credibility and respect, people start imitating some of the things I do frequently. An easy example to spot is speech patterns, but it extends to other things as well. The changes are small and scattered, but noticeable.

**Change Tactic #13: Lead by example.** People naturally imitate those they respect. Lead by example, and in the beginning be prepared to do all the work for all of your suggestions by yourself.

### 4.3. Be Careful with Vocabulary

Language is a big part of causing change. Although I led by example where possible, most of my efforts at change involved simple talking. One thing that I noticed is that my use of words didn't always correspond to the same words used by Aardvark. For example, I've always thought of a technical lead as someone who leads a project team, sets methodology, tracks schedules, etc. But at Aardvark, the technical lead is more of a business analyst combined with a programmer. There were similar slight differences in terminology across the board. This caused people to misunderstand me as well, and created some long-term confusion.

**Change Tactic #14: Be careful with vocabulary.** Jargon that means something positive to you might mean something entirely different, even negative, to your audience. For example, "iteration" might mean "continual refinement" to you and "rework due to poor planning" to your audience.

### 4.4. Create Memes

I thought of my work as spreading ideas. I concentrated on doing things that would make small changes to the way people thought. My assumption was that some of those things would spread, and if someone heard the same basic idea from multiple sources, they would give it more credibility.

I tried to talk about my ideas with everyone that I had regular contact with, and I repeated the same basic idea over and over, in different guises, until it started to sink in. It took several weeks for small, simple changes such as the note cards to sink in, and months for bigger ones.

I didn't spend every waking minute talking about problems and solutions. That would have been annoying. Instead, I just spoke up when a good opportunity presented itself. One of my big concerns was that I would irritate people by being too much of a know-it-all. So I tried to keep my comments to appropriate times. When something wasn't working or was causing pain, and it was related to a change I wanted to make, I would speak up and say "If we did such-and-such, it might help thus-and-so."

As part of this strategy, I also wrote essays. The essays targeted bigger, longer term problems and were meant for management on the fringe of my sphere of influence. I wanted to have plenty of separate vectors for my ideas.

As an example, one of the first problems I noticed was communications difficulties between team members. I started commenting on this issue nearly from the start, and I wrote an essay about the importance of communication and the value of bullpens in facilitating communication. I can't say for sure if those efforts had any result or not, but several months later, as part of a larger move, all of the cubicles were reconfigured as bullpens.

**Change Tactic #15: Create memes.** You can cause change on a bottom-up basis, but it will be difficult and impermanent. Simultaneously work on creating memes that change people's way of thinking and permeate the organization. Your bottom-up efforts will contribute to the memes, so don't get discouraged.

### 4.5. Understand Why

It was easy for me, coming into the organization from the outside, to see it as a mass of problems. I had to continually fight that inclination, because it caused me to lose respect for people, and that showed in my attitude. The longer I worked at Aardvark, the more I came to understand why things were done they were. Sometimes silly things were done out of inertia, but even then they once had reasons behind them. The more I understood those underlying reasons, the more effective I was. It gave me an appreciation for other people's point of view and they paid more attention when they felt like I really understood their situation.

**Change Tactic #16: Understand why.** There's a reason why things are done the way they are, and it isn't because your coworkers are incompetent or malicious. Sometimes they may be ignorant, but even that isn't the *only* reason.

### 4.6. Be Flexible

My preferred methodology is Extreme Programming, which I've found to be most effective when practiced with rigorous adherence to its practices. Unfortunately, that simply wasn't an option at Aardvark. If I had waited to introduce a new methodology until I could do full-blown XP, I'd still be waiting.

**Change Tactic #17: Be flexible.** Sometimes the organization needs to change to fit the process, and sometimes the process needs to change to fit the organization. It's easier to change the process.

### 4.7. Don't Gloat

I didn't get much recognition that I was causing change at Aardvark. I was given increasingly greater responsibilities, but I think the true extent of my effort was unknown.

The self-imposed silence was pretty frustrating at times. I put a lot of effort in, I had some notable successes, but I couldn't say anything. Although this was frustrating, I think it was worth it. I think that people have to believe an idea was really their own in order to change.

**Change Tactic #18: Don't gloat.** If you make a change, and it's successful, don't gloat. Don't remind people it was your idea. Don't say anything at all.

## 5. Beware What You Wish For

Successful change comes on many levels. In the first week that I worked at Aardvark, I caused some change by moving my desk to my teammates. Within a few months, I made changes to the small four-person team I was part of. Four to five months later, I took the lead of a larger team and instituted an agile process.

After I took the lead of that agile project, my understanding of the company and its client grew immensely. I realized, too late, that I had undertaken a project with a schedule that was too small, a client that didn't want to compromise, and a company that didn't support its developers.

It eventually turned out all right, but I spent several months feeling like I had gotten myself into a bad position and I couldn't ethically get out. After all, I had pushed and pushed for the changes, and now I couldn't just up and leave. Eventually, I started to feel the management support that I had been concerned about, and the situation got better. But it was scary. That leads me to my first, and last, tactic for change:

**Change Tactic #0: Question your motives.** Why do you want to change your organization? Are you just fond of a particular way of working? Think about what will happen if your changes succeed, but only partially, and you're responsible. Is it worth it? Or would you be better off changing your organization by finding a better job?

Good luck!

## 6. Acknowledgements

Before starting the job outlined in this paper, I reread Gerald Weinberg's The Secrets of Consulting [8]. It was an invaluable aid and I applied its advice daily. Many of the tactics listed here are inspired by Weinberg's tips from that book.

Other important influences were the lively discussions on the Extreme Programming mailing list [9] and the treasure trove of information on Ward Cunningham's Wiki-Wiki Web [10].

And, of course, a lot of this is common sense. I learned that from my parents.

## 7. References

- [1] M. Fowler, "Is analysis dead?", Invited talk at XP2001, May 2001.
- [2] J. Little, "Change Your Organization Diary", Online at <http://c2.com/cgi/wiki?ChangeYourOrganizationDiary>
- [3] J. Little, "Change Your Organization Tactics", Online at <http://c2.com/cgi/wiki?ChangeYourOrganizationTactics>
- [4] M. L. Manns and L. Rising, *Fear Less and other patterns for introducing new ideas into organizations*, Addison-Wesley Longman Inc., Reading, MA, forthcoming.
- [5] J. Panici, et al, "They Will Not Listen", Online at <http://c2.com/cgi/wiki?TheyWillNotListen>
- [6] K. Beck, *Extreme Programming Explained*, Addison-Wesley Longman, Inc., Reading, MA, 2000.
- [7] S. E. Heiman, et al, *The New Strategic Selling: The Unique Sales System Proven Successful by the World's Best Companies*, Warner Books, New York, NY, 1998.
- [8] G. M. Weinberg, *The Secrets of Consulting*, Dorset House Publishing, New York, NY, 1985.
- [9] Extreme Programming mailing list. Online at <http://groups.yahoo.com/group/extremeprogramming/>
- [10] Wiki-Wiki Web. Online at <http://c2.com/cgi/wiki>