

The Decline and Fall of High-Tech Corporate Culture

Nancy Ross

My friends hate their jobs. No, these aren't my flaky friends who are never happy with the circumstances of their lives. These are the solid citizens, always looking for how to do their work better and live happier lives. Still, without exception, they report being unhappy in their jobs.

Spurred by this universal dislike or even hatred of their jobs, many people in software development and test positions change jobs or companies almost annually. Companies suffer as a consequence because the already huge costs of recruiting, hiring, and retraining are growing every month. People suffer because this misery is an enormous drain on their individual resources.

What causes this rapid turnover? Are there cures? What has changed in high-tech corporate culture that leads to mass discontent, and what can we do about it? To answer these questions, I offer my perspective, having worked in a variety of software development, project management, and executive positions for the last 20 years.

So Much Money, So Much Pain

First, let's examine the apparent sources of this unhappiness. Jason worked at a Silicon Valley startup, where his peers were hostile and contentious. His boss was a young woman who didn't seem to value his maturity and experience. After vesting some of his stock options with this company, Jason decided he couldn't take it any more and moved on to another startup that seemed to

have values and practices more in line with his own. But, after six months at this new company, neither the position nor the company seemed to be going anywhere, so he contacted the previous company about the possibility of returning. They offered Jason a job, but then quickly offered him a wad of cash to *decline* the job after all. After some investigation, they had found that although some people at the company wanted him to return, others did not. The best solution, they concluded, was to retract the job offer, stanching Jason's pain of rejection with a large sum of money for not working there. It's almost as if a person could make a living in Silicon Valley by turning down jobs!

Marsha works for one of the granddaddy Silicon Valley firms that spawned many high-tech businesses. Her boss, younger than Marsha, seems to work at cross-purposes with her. For example, Marsha's boss doesn't adequately communicate with her, informing her at the last minute of meetings with partner companies or leaving her out of these meetings altogether. Instead of working collaboratively, or dividing their work based on interests and abilities, they're enmeshed in a power struggle that denies the company the full benefit of both their skill sets. Marsha is now thinking of working for the company down the street, where she *might* not face the same daily struggle with her manager.

Connor, who worked at a middle-aged Silicon Valley firm, had a blaming, unsupportive boss—and peers to match. As the head of a software test team, he instituted the use of metrics to track product quality, the number



and severity of defects, convergence toward ship date, and so forth. Rather than getting pats on the back for bringing these sound practices to the business, he was publicly berated when he shared this information in meetings. After enduring enough of these public humiliations, he went to work for another well-established company in the Valley, where he has had a new manager about once every three months. Now, he's having trouble getting a solid set of objectives and performance feedback.

After Andrew took a leave of absence caused by the extreme stress of his previous position at a strong up-and-coming Silicon Valley firm, he returned to the company in a position that his boss asked him to fill. This was a role previously filled by the boss himself. Andrew now finds himself "invisible" in the position, with his boss apparently unwilling to give up the role he's asked Andrew to fill, while holding Andrew responsible for the outcome. Andrew would like to leave the company, but the potential upside of the stock options he's been granted makes the misery worthwhile, at least for the time being.

The common complaint in the work lives of Jason, Marsha, Connor, and Andrew is the immature, hostile, passive-aggressive, or (insert your adjective here) boss and coworkers. Although Silicon Valley might be the most extreme case as far as frequency of job changes, I believe this job-related misery is predominant in the software industry throughout the country—a situation that has become worse over the years. Let's look at what has changed.

How We Got in This Fix

First, the huge demand for skilled software workers lets people change jobs and companies at will, thus relieving the need for employees to find ways to work together in harmony. If it doesn't work out, we move onto

the next unhappy marriage, finding temporary relief until the thoughts of divorce invade again. This short-term



Sally Lee

thinking affects not only our own approach, but also that of our bosses, who, more often than not, seem to want to be the most valuable player of the big game, rather than to play with the same team next year. Short-term jobs are not the black mark on a resume that they once were, further encouraging us to move on rather than work out problems.

Second, rewards are often based on individual, rather than team, work. We tend to reward the golden boys and girls for their moments in the sun, rather than for sustained performance. In moving toward new compensation models, such as a lower base salary and higher variable pay based on individual performance, we exacerbate this situation. Although it is possible and sometimes desirable to use this compensation model to drive team performance, as companies and managers, we frequently foster an unhealthy competition among coworkers with such an approach.

There is another very important compensation-driven reason for short-term thinking. The biggest financial rewards of all come when an employee

changes companies. Signing bonuses and other up-front compensation can double annualized rewards. Because this is far more compensation than a person receives by staying in a job, there is a strong incentive to get some quick glory to add to the resume and then jump jobs again, bumping up base salary by 50% or more.

Stock options, on the other hand, throw a nasty curve into this otherwise painfully clear compensation game. Because options have vesting schedules from one to four years after granted, they are the famed golden handcuffs. Without them, software workers would change jobs even more often. With them, employees stay in positions and companies that no longer suit them. Imagine the misery these people spread throughout the organization as they resentfully do their time!

When the Sex Is Good

Why is there such a huge emphasis on compensation issues, you might ask? We hear it repeated, almost as a mantra, that the primary motivator for people in the workplace is not compensation but job satisfaction, recognition, and other "soft" factors. I believe this is true. However, what is happening is analogous to how a friend once described the importance of sex in a relationship: "When sex is good, it's 10% of the relationship, and when it's bad, it becomes 90%." A very similar phenomenon is occurring among software workers. When a job is satisfying, compensation is only a 10% concern, and as job dissatisfaction grows, the focus on compensation issues increases, eventually eclipsing the more basic needs for recognition and satisfaction on the job.

Insecure or immature bosses are another driver of miserable workplaces. Bosses might be insecure because they have outlived their usefulness in a given position or company,

but are afraid to move on. They stay in a job, sometimes far past their expiration date. The fear they feel plays itself out in control games, power plays, thrashing, and any number of other unproductive behaviors. People who work with insecure managers are often confused and frustrated.

There are many immature bosses in Silicon Valley, due to the shortage of experienced managers, compounded by senior management's blind hope that inexperienced managers will do more good than harm. Immaturity might come in several forms. Some managers are just young and don't yet have the maturity and compassion that often come with age. These brash youngsters cut their teeth on the rest of us, who are left to nurse our wounds until the next gash is inflicted.

Another type of immature manager has insufficient relevant experience at the level of management in which he or she is attempting to function. These functionally immature managers do as much or more damage than their age-immature counterparts, because they're often doing it at a higher level of management, thus affecting more people.

The Courage to Change

How can we create a more worker-friendly environment and even reintroduce the notion of career to the software workplace?

First, companies need to set appropriate standards for professional behavior and bring it to people's attention when they operate outside these standards. Particular standards might vary from organization to organization, but the basics of respectful treatment of others, cooperation, and open yet courteous communication can form the basis for these standards of behavior in any situation. I worked at a company that strictly adhered to standards of professional behavior, where management promptly corrected those who stepped outside the bounds. This company has a reputation far and wide as an excellent workplace. At

the time, I sometimes found this correcting behavior confining; I have since come to appreciate it as a key factor in making the company an excellent workplace.

Next, middle and upper management need to value maturity—both personal maturity and maturity within a job function—in candidates when determining who is a good fit for a given position, especially in management. Too often, a middle manager determines whom he or she wants to fill an open management position, then asks stakeholders to interview this candidate for the position. Regardless of the feedback the hiring

Far too often, problems are ignored or ineffectively dealt with until they reach a fever pitch, and the people involved have become sick, tired, and ready to move on.

manager receives, the candidate gets the position. For example, we interviewed one candidate, a friend of the hiring manager. We reported that the candidate could not answer a single question solidly, indicating a lack of ability to do the work. The hiring manager glossed over the feedback and promoted the candidate anyway. Not surprisingly, the individual was a disaster in this position, for which she wasn't prepared. A rash of resignations and transfers out of her group followed her promotion. The hiring manager took the expedient path for the promotion, which in the long term caused huge problems in the organization. To *avoid* this requires tremendous discipline, judgment, and courage on the part of the hiring manager.

This same discipline, judgment,

and courage on the part of managers are critical in a wide variety of situations they must address. The situations in which they must use these qualities will run the gamut and be unpredictable in nature and timing. Effective managers constantly look for situations such as those in the examples I've cited, before and as they develop. Ideally, we would catch all such situations before they become problems; in reality, many will become real problems before we recognize them. Once recognized, we must have the courage to assess the situation and deal with it. Far too often, we ignore or deal ineffectively with problems until they reach a fever pitch, and the people involved have become sick, tired, and ready to move on. Dealing with really tough environment and people issues is the most important part of our jobs as managers.

On the compensation front, look for ways to reward cooperation and teamwork that help achieve project goals. Reward individual contributions as well, but only when there's no negative impact on the team or on the project goals. Set project goals that encourage people to cooperate to reach team goals, rather than setting individual goals that can be attained at a cost to the overall project. Also, be sure to reward for sustained performance, thus setting the expectation that your employees will work together productively over the course of a project's lifetime. You should also recognize attaining interim goals, which helps maintain morale and sets the stage for the project's successful completion.

Finally, the situation that perhaps requires the most courage by a manager is the one in which it's necessary to encourage an individual to move on, either to a different position or to a different company. People who stay beyond the point where they're happy and productive in a position become destructive influences to themselves, the organization, or both. Do everyone involved a favor and discuss such people's future options with them.

None of these suggestions are easy to carry out. They require a management discipline that we must build into our organizations. It's also possible, of course, for the badly managed to take the first step.

For example, if you're working in a job that's dissatisfying, consider how to broach the subject with your manager. It's not effective to say, "You're a lousy manager." What could be useful is suggesting something the manager can do. In Marsha's case, she could say, "I'd like you to consider when I need to be invited to meetings and when not. Let's talk about it and devise a strategy." Marsha might offer to save her boss time in meetings as a way to influence her boss to do the right thing.

Connor could define his objectives and write his own job description. Chances are good that the company would be thrilled he showed this much initiative.

Andrew could decide to stay for the money, in addition to negotiating with his boss on specific actions that he, Andrew, can take to be successful in his job.

Start by taking a few small steps to correct the problems in your organization that you might have identified as you read this column. In the long run, the organization that provides the best work environment will attract and retain the best people. Having the best people productively work in our companies is what will make our products, services, and workplace stand out from the competitors, and revenue growth will reflect this. This is what success in the marketplace is really all about. ☺

Acknowledgment

I thank Johanna Rothman and Jerry Weinberg for their invaluable editing assistance.

Nancy Ross just resigned as director of software engineering at Becton Dickinson in San Jose, California. Her research interests include software development, test management, and best practices. She has a BA in computer science from Indiana University and an MS in computer information and control engineering from the University of Michigan. Contact her at nancy_ross10@hotmail.com.

SOFTSTAR
ESTIMATION EXPERTS

On Target Estimates
FAST AND EASY!

COSTAR 6.0

Costar 6.0 is an easy-to-use software estimation application for Windows that makes it easy to produce accurate and reliable estimates of project costs, duration, and staffing.

Based upon the latest version of the industry-standard COCOMO II estimating model, Costar makes it easier to estimate and plan software development projects. In addition to supporting traditional project development models, Costar supports Incremental COCOMO, allowing you to model development as a series of concurrent software projects, each yielding an intermediate product.

You can create alternative project assumptions, and compare the results side-by-side. You can ask "what if" questions, such as "What if we postpone a particular feature until the next release?" or "What if we had more experienced software developers?": Costar even lets you constrain your project to a fixed schedule or a fixed budget.

- COCOMO II.2000 calibration
- REVIC model
- Easy export to Excel
- New tree view for component navigation
- 16 reports & graphs
- Enhanced help system
- MBASE, Waterfall, or REVIC phase models
- Up to 20 increments per project
- Function Points
- Full documentation, including:
 - Software Engineering Economics
 - Software Cost Estimation with COCOMO II
- Site, Network, and Corporate licenses available
- Free unlimited support

GET FREE CONSULTATION
603-672-0987

Download your free demo copy of Costar 6.0. After installation, call us for a free 20 minute consultation. We'll get you started right.

Buy Costar 6.0 and you'll get free **unlimited support** and estimating expertise.

Download your free demo copy at
www.SoftstarSystems.com/ontime