



Introducing Agile Without Trauma: A Pragmatic Approach

Bruce Taylor, Senior Agile Coach
ManagingProgrammers.com

All through the world of software development, companies are deciding to “go agile.” In many cases this decision has been well thought out, but often the organization has more of a wish than a plan. A good agile coach must be able to step into either situation and help the company move toward agile *as much as is possible at the time*. Obviously, there is no recipe for causing this transformation, or there would be no market for agile coaches. The rest of this paper is a meditation on what is possible, what is desirable, and how I approach the uncertain world of agile migration.

Necessary Conditions

I’ll oversimplify for a moment and assert that three conditions must exist for an agile transformation to be feasible:

- Corporate executives and upper management must understand the company-wide risks and benefits of “going agile,” and must be strongly committed.
- Middle managers must honestly feel that they have freedom to experiment and take risks during the transition, and that honest mistakes will not be punished.
- Team leaders and individual contributors must understand the new freedoms and responsibilities of going agile, and must show a high degree of flexibility and integrity.

I hasten to say that these conditions are *necessary, but not sufficient* to ensure success in the agile transition - there are many other factors that can interfere with the transition. I also assert that these conditions are not binary; an organization can be more or less ready at any of the three levels. And finally, an organization’s readiness on these conditions determines how much transformation is practical at this moment.

Phase 1: Assessment

When I begin an engagement to help a company transition to agile development, I spend several days simply gathering data on the organization’s readiness. Typically, I will interview as many of the executives and upper managers as I can, talking with them about their expectations for converting to agile: how much risk they are willing to take and what benefits they expect. I spend a lot of time talking to and attending meetings with middle managers, making contributions where I can, and gauging their approach to development: how much they will risk for the possibility of significant benefit. Then, I spend as much time as I can with the programmers, testers, DBAs, product managers, and all the rest of the people who actually build the product. I talk to them about their experiences and aspirations, what gets in their way and what they need to be more productive. And finally, I read a lot of technical documentation and code to get a feel for the level of technical debt in the product. In short, I gather a tremendous amount of information before I make my first recommendation, and I try to gauge the company’s readiness by scoring it on the big three questions of the previous section.

Phase 2: Planning

I try very hard never to surprise anyone - and especially not the executives and upper management - so I put together a draft plan with at least three recommendations: the amount of change to attempt at this time, the strategy for introducing the change, and the timeframe for introducing agile to the organization. My recommendations may range from a “baby step” for companies that seem very rigid and set in their ways to “let’s just do it” for a company that is committed to the change from top to bottom; but it is always possible to introduce *some* level of change, then come back for more when that has settled in.

Phase 3: Engagement

I am very committed to the coaching model of change management. That is, I never tell anyone to do anything, and I very rarely even give advice. Instead, I provide training in agile techniques that others have found helpful, and assist the managers to choose the ones that match their own organization. I help them integrate the techniques into a development process that may contain elements of Scrum, XP, or even waterfall - whatever seems right for this organization at this stage of its evolution. By getting the managers to make the decisions and structure the process, I make them owners and ensure that they have a personal stake in its success.

I do the same thing with the programmers, though I can’t provide the same one-on-one coaching that I give the managers. But I give them training in best practices and software quality techniques, and help them decide what tools, training, and processes they need to be more productive and flexible. Since programmers are remarkably ingenious, their ideas often filter up to influence the process that the managers are constructing.

When I interact with the executives and senior management I provide a mixture of assessment and reassurance, because it’s remarkably hard to see an agile organization forming if you’re not an intimate part of the process. So I keep them apprised of the current state of change and encourage them to look ahead to the next step in becoming agile.

Phase 4: Followup

Basic psychology teaches that behavioral changes fade quickly if they aren’t reinforced regularly, and the same holds true for organizational changes. So in the final part of the engagement I visit the organization regularly to look for things going awry, or to help managers recognize things that are working well. I provide encouragement where it is necessary, because any change can be disheartening, and I try to keep the momentum going. And at some point, I go back to the upper management to start planning the next step in the agile transition.

About Bruce Taylor



Bruce Taylor is the principal of ManagingProgrammers.com, a management coaching firm located near Boston, Massachusetts. Bruce helps software organizations of all sizes to create low-stress, supportive, adaptable working environments, so that the engineers, leaders, and managers can work as effectively as possible. He provides executive coaching for senior managers who are creating superior organizations, management coaching for technical leaders who are adapting to new agile practices, and individual coaching for engineers who are upgrading their skills. Bruce has a Masters in Computer Science from Duke University, a Masters in Community Psychology, and a Certificate in Job Stress and Healthy Workplace Design, both from the University of Massachusetts.