

CHAPTER TWO

Do the Right Thing

To genuinely improve our collective ability to create great software products, we need to recognize that the critical components of getting better have been largely ignored. A sustainable solution begins within each of us.

In this chapter, we make a distinction here between the approaches we normally take for getting things done and a different approach where we all proactively drive teamwork and cooperation. If we all step up and work together toward a common goal, we are much further ahead.

Doing Things Right vs. Doing the Right Thing

A distinction needs to be made in the software world, one that can generate much more effective use of our valuable time and resources.

Often, improvement initiatives are based on the desire to become more technically efficient. This is the “silver-bullet” solution and is often tools based in nature. If effectively managed, these approaches can indeed generate productivity gains or higher predictability in getting the project completed as expected. This approach is the *doing things right* method. We ensure that, given a task, we do it in the most expedient manner at our disposal. Doing things right is always apparently urgent, but rarely critical. This approach generates relatively easy solutions with relatively small gains.

However, to truly maximize the value of improvement initiatives, you need to *do the right things*. This means working on the root issues that result in the greatest benefit.

At the business level, it is important to ensure that you meet the client needs, not just get the current project done. It has to be the right project. There has to be a strong business case for proceeding with any project; and as the shape of the project changes over time, the case driving that project needs to be revisited, too. To stop a project that no longer has a viable business case is not failure. It makes good business sense to divert those resources elsewhere.

At the development level, there are an overwhelming number of tasks that you could do right. If you try to do them all with your limited resources, you are sure to fail on both time and cost factors. A key part of managing development activities is the appropriate selection of the right activities that will most likely lead to overall success. Focus must be placed on the work that provides the best value for reducing uncertainty or generating part of the final product.

We need to strive for effectiveness over mere efficiency. At all levels, it is often a safe bet that the appropriate tasks to tackle are the ones that are closest to the root cause of the problem. They will have the greatest positive impact on success.

Sustainability

When we talk about sustainability for software businesses, everyone brings a very different perspective to the table.

There are those who would be happy to make it through the day without significant grief and disruption. These individuals often work in an environment where each new day brings a different set of surprises and challenges, often negative. For these people, sustainability means reactive survival, and there's no point even looking at a more distant horizon.

Others can see past the daily grind. Sustainability for them is measured by project completion, which may be weeks or months out, and drives daily activities. This longer context brings greater meaning to their daily work.

We start thinking strategically when we consider the parade of projects that will sustain the organization over a longer term, but even here there are shades of gray. There are times when what has to be cut from the current release drives projects, a tactical rather than strategic approach. For companies accountable to shareholders, strategy might mean only planning as far out as the next quarter, or planning for the anticipated IPO or acquisition.

None of these appear to be sustainable models. With new companies sprouting up whose objective is to be acquired as soon as possible (ruthlessly driving for higher sales figures rather than sound financials), with CEOs specializing in quick turnarounds that reflect positively on the current quarter at the expense of downstream results, with VCs focusing on their own IRR and liquidity events, identifying the companies that truly emphasize long-term sustainability becomes even more difficult.

At the executive level, it is rare to see an appreciation for true sustainability that goes beyond fiscal reward. Success is measured primarily in the packages negotiated and the toys accumulated, driven almost exclusively by tactical approaches. Indeed, those driving a sustainable business (insert your truly visionary executive here) stand out because they are so rare in today's world. It is difficult to focus past the tactical milestones. It requires a concerted effort from a well-coordinated team. Unfortunately, single-mindedness is often interpreted as "it's my way or the highway."

Below the executive level, most people aren't normally expected to consider long-term sustainability. They're the worker bees who get the job done. Although many at this level appear to be comfortable with such an Orwellian approach (at least initially in their careers), there is significant value and reward in contemplating and focusing on the bigger picture. Indeed, many of us need to understand the role we play in the larger context to become a true stakeholder in the shared success with the company. As we grow, we gain an appreciation for alignment with the overall vision. Those who are unable to gain that perspective in their current organization

will eventually move on, and in doing so, impact the sustainability of the organization they are leaving.

At all levels, the key ingredient for sustainability is active involvement from the people who make up the organization. Although participation does not necessarily mean consensus (this appears to work only with a relatively small group), it does mean that there is honest and open engagement at all levels within the organization. There needs to be an appreciation that this investment in human capital is essential, rather than perceiving human resources as a manageable expense. With a truly long-term vision of what the business intends to provide to its clients, the entire group can align themselves and work miracles.

Goals are better achieved when everyone is synchronized and willing to collaborate on a strategic approach.

The funny thing is, a focus on building a truly sustainable business is not necessarily at odds with any of the more tactical goals driving many of today's companies, even if it may be perceived that way. This might be more difficult, but it is also much more rewarding, especially if success is measured in terms beyond a paycheck.

The Way Things Are Done

Rear Admiral Grace Hopper said, "The most damaging phrase in the English language is 'It's always been done that way,'" and I tend to agree.

We all resist change, even if we know deep down inside that it would be for the best, and no shortage of rationalization takes place to help us maintain the status quo. We invent all manner of excuses for preferring the current situation to any change, and when all else fails, we fall back on the old standard.

It is important to recognize that change comes in many shapes and sizes, from those changes that are essential for our survival to the changes that will

set us back several steps, from trivial tweaks to overwhelming disruptions. We need to train our minds to recognize opportunities where a change can add value, instead of simply refusing to take on any changes at all.

If you ever run into a situation where you can't possibly comprehend the value of the task you are being asked to do, ask around. There may be a chance that the rationale for performing a task is truly valid, but just not clear on the surface. If nobody can give you a valid justification for doing the work, there is a significant problem here that needs to be addressed. In most cases, it makes sense to focus your efforts on tasks that more clearly contribute to the organization's goals.

Part of the Standard Agreement

I was involved in a discussion with a potential referral partner a few weeks ago and had some concerns with the restrictions in their "standard" agreement.

They indicated they couldn't drop these restrictions, despite my suggestion that they were not in the best interest of our collective clients. Their only argument for their case was that they've always done it that way.

Their argument made no sense to me, and certainly provided me with no benefit. They are now an "ex-potential referral partner." I would rather be able to do what I need to so that I can serve the needs of my clients in an unfettered manner.

If the organization refuses to act on evidence of dysfunction—if you can't find out why you are being asked to do something—perhaps the problem runs deeper. At some places, all manner of tasks are performed simply because people have been told to do them, with no regard to the bigger picture. The internal cost of this overhead can be huge, and the human cost of not knowing where you contribute to the overall perspective can be just as devastating.

Questioning activities needs to be balanced against the potential for inflicting too much change too quickly in an organization. When a significant amount of effort is wasted on activities that don't clearly contribute to the overall vision, be careful not to introduce too much change. The associated hysteresis (and resulting hysteria) can be as strongly negative as the original undirected activities. Pick a few key changes, make the adjustments, assess the results, and then pick the next key areas. There will always be areas to tweak. Change is not a one-stop proposition.

When developing or adjusting a defined approach for your organization, always consider why you are taking the approach you have chosen. There should be sound rationale for each step of the process, beyond rationalizations such as "that's what we've always done," or "that's the standard for this process," or "it was easy to do it this way."

A significant part of any defined approach should be to provide clear justification of the rationale for each of the steps, to identify the situations where the suggested approach will and won't work, and which issues may arise. Even if you aren't in an organization where people will ask these very important questions, how could you possibly justify deploying something that hasn't been validated against these benchmarks?

Learn to be comfortable with an appropriate amount of ongoing change. Accept this change, seek to continuously renew, and avoid the standard rationalizations for maintaining the status quo.

We Need to Take Ownership for Success

We've all worked in situations that are less than optimal in this industry: projects that seem to go on forever, difficult people, long hours, constant surprises and fires to put out, the feeling that nothing will ever get done. In many organizations, the situation degrades to a point where those who

could potentially see the big picture get their feet swept out from under them. The problems that appear to be endemic to your company are actually epidemic throughout the field.

In a culture in which missed commitments are the norm, it is easy for those with even the best intentions to ease into a comfort zone of blame. Sales and marketing seem to promise the world. The development group can't seem to meet expectations that were set elsewhere. That test team just can't be satisfied. Management wonders whether they will ever be able to herd all these groups to closure, and the customer can't seem to make up his mind. It's easy to place blame on one or more of the other groups, because you can then absolve yourself of any responsibility for change—it's outside your sphere of influence.

That view is shortsighted. All these groups with different responsibilities need to coordinate their efforts to bring the project to closure. This coordination is a combination of individual efforts toward a common goal. Realistic commitments need to be agreed upon and achieved. When they are not achieved, the root cause needs to be resolved by the group without resorting to blame. If the system is looked on as a whole rather than as a collection of separate teams, it is rare for external causes to be the cause of failure. The problems lie within, and are more often within our sphere of influence than we care to admit.

Software development is done by people, and is driven by relationships and communication. For any endeavor, effective communication is the primary asset that differentiates the successful team from the challenged group. To succeed, there needs to be an acknowledgement of individual contributions and clear intent to communicate from all individuals. We are all stakeholders with significant influence.

Owning Our Future

Whether we are officially in a recession or a full-blown depression, whether or not the improving economic indicators are real in the face of continuing challenges in the world, it is unnerving to observe how many executives will abdicate responsibility for their company's performance.

Tossing Ownership Aside

A survey across one entire organization revealed a fascinating indicator of ownership in action.

One question asked the entire group to identify their key sources of pain. Although there were a few canned elements they could select, there was also an “other” category where they could enter free-form text.

All the technical staff placed responsibility for their woes on internal issues, such as inadequate requirements analysis or insufficient testing.

Every response from the management team suggested external issues contributed to their pains. The downturn in the economy, the weakening exchange rate, and the recent war were all cited as major issues.

This was a management team in serious trouble but unwilling to step up and acknowledge there were elements within their control to drive change.

We seem to be facing a growing range of external threats such as wars, diseases, and severe weather, keeping uncertainty high and the economy challenged. With all this happening, it is unnerving to see how many companies just lay low in these tough times, waiting for external circumstances to change so that they can come out of their bunkers and flourish again. A couple of years after the dotcom implosion, we started to ask, “So, do you think things are turning around?” as an icebreaker at networking events. Usually, the response was “Things are looking better...,” but the inflection was that of another question rather than an indicator of confidence.

My bet is that the external pressures driving uncertainty will not fade. There will always be conditions and events that make it difficult for businesses to easily thrive. After the dotcom bubble of huge returns and short-lived successes, when we finally realized that the “new business model” was unsustainable, we now remember that driving a successful business is not a walk in the park. The new business reality is that there is

no new business reality, and likely will not be anytime soon. The fundamentals remain the same.

The best way to gain security for your future is to take responsibility. You need to acknowledge that the results you reap will be primarily influenced by your own actions. To look elsewhere is to search for excuses or blame, not solutions.

Carpe Diem

Seize the day. A literal translation from Latin is “to grab the current opportunity with gusto, without regard to the downstream ramifications.” We can be more proactive at making these opportunities occur, instead of waiting for them to come up and bite us on the backside.

You can follow a few steps to take control of this day. First, you need to know what you are looking for in that day. Having clear, specific goals is critical in software projects and in life. Although I usually despise corny acronyms, SMART (specific, measurable, achievable, realistic, and tangible) characteristics are critical attributes of good goals. These are the strategic targets that describe the positive future that you are working toward. “Begin with the end in mind,” as Stephen Covey counsels.

When you understand your goals, you can then identify what these opportunities look like so that you can recognize them when they appear. Is it a new technology that you can leverage to your advantage? Maybe an unexpected opportunity for reuse arises that may cost you a little time now but save a bundle later. Perhaps an unlucky break somewhere else takes some of the pressure off of your deadlines. That one satisfied customer might be able to point you toward two more.

What seem to be just positive events can induce more strategic consequences, assuming that you understand where you want to go. The first one to jump on that new technology can become an industry leader. That extra time saved can allow you to focus on a new initiative that turns you in a bright new direction. One of those new prospects can turn into a strategic partner. Things start to happen.

You can also take this one step further, by proactively seeking these opportunities. With the recognition that each of these leads has the potential to become much more than just a nice event, the motivation to take charge of your own future increases dramatically. What happens is that you are always on your toes, watching for the opportunities, making these opportunities, and ready to run with them as soon as they arise.

When taking a planned, strategic approach, we need to go beyond *carpe diem*. We do not just seize the day opportunistically, we do not rely on luck or circumstance, we are in control. It is more a matter of *procreo diem*. We create our days and thus the outcomes we desire.

A Solution Framework

The solution space we have traditionally inhabited is too narrow. Tools, frameworks, and methodologies are insufficient. We need to add the elements of alternative perspectives and respect for the human condition, along with transparent communication, to the mix to identify lasting solutions that work for everyone.

Figure 2-1 presents one way to look at the range of issues of the software development space.

As with any human endeavor, software development starts with individuals. Each has his or her own unique set of values, motives, attitudes, and skills based on experience, training, and aptitude. Everyone else's personal environment is just as real as yours. Any affront to that environment, any lack of appreciation for "where you are coming from," is not taken well. Emotions are far more tactical, but just as real and important to consider as the longer-range motives and attitudes.

When these individuals gather into groups, we start to deal with interaction and the formation of relationships. We've all been involved with relationships that have been effective and with those that could stand improvement. The goal is to move into relationships that are positive, that embody mutual respect and a sense of belonging.

When organized into teams, we then are dealing with the coordination of a diverse set of individuals. Teams lean on systems and guidance to help

Overall success for stakeholders is dependent upon successful management of individuals into groups and groups into teams – this should not be left to chance.

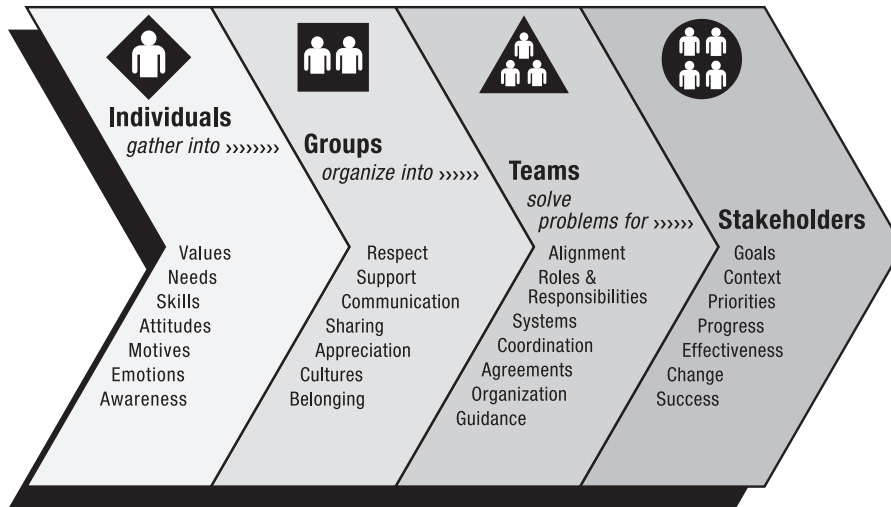


FIGURE 2-1

The range of issues in software development

them align toward a common vision. This is the first step teams take to solve a problem.

With appropriate direction, these teams will work together to solve problems for stakeholders. Not only have they evolved systems for interaction, but they are now working toward a specific common goal. The intent is to progress toward that goal and eventually an ability to determine whether the endeavor was a success or a failure.

This progression is most effective when taken in order, with nothing implicitly left to chance. Each piece taken for granted is an invitation to failure, particularly if success and failure is measured in dimensions beyond having shipped on the target date.

When looking at the breadth of issues across this range, from individuals through to stakeholders, typical approaches such as the incorporation of a packaged process framework or the purchase of a tool tend to provide superficial support at best. Although they can be seen as critical for structuring and managing huge amounts of data on large projects, all tools depend on

appropriate and accurate data for good results. For requirements or design tools and for packaged processes, the overall results will be no better than the quality of the information used as input. This data and information comes from interactions across the team.

Tools, frameworks, and methodologies tend to emphasize those aspects of organization that are in the Teams area in this figure. These are necessary, to be sure, but hardly sufficient. Have you ever been involved in the deployment of a tool or methodology without 100 percent buy-in? Many purchased solutions of this type are no longer used even a year after purchase. It is the team that chooses whether to use the tool, or to learn how to use it effectively.

Our ongoing reliance on tools and methodologies is misguided in that it is largely prescriptive. The issue is not the original intent, but how this intent is translated into practice by most teams. A superficial understanding of the rationale and applicability behind well thought-out approaches drives most teams to implement these changes in a dysfunctional manner.

It is all about the people.

Software Is a Team Sport

Our motives, attitudes, skills, and team relationships are all fair game to explore and leverage as opportunities for change. We need to take personal ownership and responsibility for the results of our behaviors.

Effective software teams are more than a management issue. We cannot equate software development to the assembly-line floor, despite attempts to do so.

Procedures can serve as a guideline for what to do and provide structure for managing our information. However, they are insufficient to guarantee success in all but the most trivial cases. Projects and tactical situations are all different. A key component of these differences is the evolution of relationships as teams work together. If not carefully managed and nurtured, these relationships will almost surely erode over time.

Software is a creative endeavor done with real humans (with emotions, feelings, needs, and concerns that are both diverse and important). As Maslow suggests in his hierarchy of needs, we cannot rise into the ranks of optimized teams until our physiological and safety needs have been properly dealt with.

We need to take responsibility for explicitly and consciously managing our relationships in our teams, and to proactively design the approach we use as a team to develop software. This demands participation from all stakeholders involved and appreciation from all parties of the contribution (technically and emotionally) from everyone at the table.

Communication needs to be precise, open, and transparent. Hidden agendas cannot be tolerated, and we all need to be more effective with the skill of active listening. We need to be capable of empathizing with the positions of others before we pitch our case in discussions. Listening needs to be more than biding time to formulate our counterargument, more than waiting for an opportune moment to blurt out our perspective, and more than tolerating that noise from the other person's mouth.

Each person needs to respect differences within the group and be sure to address concerns from these different perspectives. *Vive la difference*. These differences make for unique and innovative solutions, long-lasting and strong relationships, and teams that will succeed now and in the future.

We all need to be able to consider the ramifications of our actions and understand the rationale behind what we are being asked to do. If something does not fit well with our mental model of appropriateness, we need to be able to speak up. Silence is not golden, and the absence of conflict is too often apathy. If the fit cannot be made, we need to be prepared to talk with our feet.

Although this approach is a tougher and deeper commitment to change, the solution is much longer lasting and significantly more rewarding. We need to stop playing with the symptoms. There really never will be a silver bullet.

Communication is a two-way street. Communication builds trust, and trust is essential in a team environment. We need to take charge, work together to build an effective team, and finally take ownership for software development.

Summary

For effective change in the workplace, we need to respect the distinction between doing things right and doing the right things.

We need to overcome the inertia of behaving in a certain way simply because that's the way it has been done, and we must recognize that we are responsible for ensuring that our actions lead us in the right direction.

To do this properly, we must first accept that we have a stake in the results and that we need to manage how we interact as a group. Then, we can apply the appropriate best practices that are reasonable for our culture and situation so that we have the best chances at success on our projects.

How Is This Relevant?

Sustainability. Try to identify the companies in your city that demonstrate true sustainability. What percentage of the overall population do they cover? How do they show that they are interested in success beyond the next quarter, or product cycle, or buyout? Do you consider them successful?

Small, personal steps. Achieving the big goals starts with achieving the small ones. How are you going to interact with the team today to ensure that you are taking ownership for success rather than blaming others for the challenges you are facing?