



WHY AGILE WORKS

The Values Behind The Results

Michael De La Maza
David Benz

InfoQ^{ueue}

ENTERPRISE SOFTWARE
DEVELOPMENT SERIES

Why Agile Works Mini-book

© 2016 Michael de la Maza & David Benz. All rights reserved.

Published by C4Media, publisher of InfoQ.com.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recoding, scanning or otherwise except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher.

Production Editor: Ana Ciobotaru

Copy Editor: Lawrence Nyveen

Cover Design: Maiez Mehdi

Interior Design: Dragos Balasoiu

Contents

1. Everything Is Agile	1
1.1 Crossing the chasm	4
1.2 Adoption vs. transformation.....	7
1.3 How the rest of the book is organized.....	10
2. Agile Values	11
2.1 Values and beliefs.....	12
2.2 Values unite, beliefs divide.....	13
2.3 Organizational culture determines results.....	16
2.4 Agile is a set of beliefs.....	21
2.5 The evolution of organizational culture.....	24
2.6 A quick digression.....	27
2.7 A values framework for agile transformation.....	28
3. Trust.....	31
3.1 Theories X and Y.....	32
3.2 Trust equals speed.....	34
3.3 Aspects of trust.....	36
3.4 Common organizational beliefs about trust.....	49
3.5 Example: How to destroy a high-performing team.....	50
4. Responsibility.....	53
4.1 Relationship to trust.....	54
4.2 Aspects of responsibility.....	58
4.3 Common organizational beliefs about responsibility.....	70
5. Learning.....	73
5.1 The world is unpredictable.....	74
5.2 Aspects of learning.....	77
5.3 Common organizational beliefs about learning.....	92

6. Collaboration.....	95
6.1 Collaboration compared to cooperation	97
6.2 The agile framework for collaboration	100
6.3 Aspects of collaboration	101
6.4 Common organizational beliefs about collaboration	112
6.5 Example: Collaboration at Pixar	113
7. Agile Values Revisited	117
7.1 Culture > process.....	118
7.2 If culture > process, why does change focus on process?	119
7.3 The agile culture and a few recommendations	120
7.4 Coda and kudos	122

PART ONE

Everything
Is Agile

“In any moment of decision, the best thing you can do is the right thing, the next best thing you can do is the wrong thing, the worst thing you can do is nothing.”

— Theodore Roosevelt

The list of companies touting agile is long.

Some of the software companies might be familiar. Spotify is agile. Salesforce is agile. Google, Apple, Amazon, Yahoo, Red Hat, Adobe, and Facebook are agile. Smaller, lesser-known software-development companies such as Atlassian, Paycor, Pivotal Labs, BNA Software, Hotels.com, and DevSpark are agile.

Companies we don't typically think of as agile are working to be agile. Microsoft, a company known for linking releases of their flagship products (Windows and Office) to specific years, claims to be agile. General Electric is agile. Hewlett-Packard is agile. Bank of America is agile. IBM is agile. Key Bank is agile. The BBC and British Telecom are agile. The United States Department of Defense is agile.

The Software Engineering Institute, originator of the Capability Maturity Model (CMM) now known as Capability Maturity Model Integration (CMMI), a top-down approach that is almost antithetical to agile, now claims that it is possible to embrace both [Gla08].

Game developers are agile. Financial companies are agile. Media companies are agile. Banks are agile. Universities are agile.

In *The Agile Mind-Set*, Gil Broza asks an intriguing question: What noun typically follows agile?

Broza writes [Bro15]:

People talk about agile development, agile project management, agile processes, agile methods, and agile best practices. Some speak about the agile methodology or the agile framework. Others refer to pairings like Scrum/agile and lean/agile.

The language of agile is everywhere.

Consultants talk about becoming agile to avoid disruption. Terms like extreme programming, Scrum, and kanban are tossed around as ways to

become agile whether people know what they mean or not. “Sprint”, “iteration”, “backlog”, and “burn down” are all entering the lexicon.



Figure 1.1: Bus-stop advertisement for agile consultants in Chicago.

National Public Radio is agile [Put14].

Forbes describes what agile leaders look like [Dut11]:

Agile leaders are not only fast and effective problem solvers when dealing with situations they’ve never dealt with before, but they are also laser-focused on results and excellent at reshaping plans and priorities when faced with unexpected changes in the environment. They are resourceful and competitive. And, they get it done fast.

Offices in Europe being designed for agility include Microsoft Netherlands, Alcatel-Lucent, Unilever Switzerland, W.L. Gore & Associates, and Eneco [Off15]. The European banking giant BNP Paribas is also agile [Sar04]. Singapore’s government is investing \$1.2 billion in technologies including agile to enhance operational efficiency and public-service delivery.

Agile certifications and assessments abound.

The Scrum Alliance, Scrum.org, the International Consortium for Agile (ICAgile), the Project Management Institute (PMI), LeanKanban University, the Scaled Agile Framework (SAFe), and the Dynamic Systems Development Method (DSDM) consortium all offer agile certifications. The In-

ternational Software Testing Qualifications Board (ISTQB) and Certified Agile Tester (CAT) offer agile testing certifications. Smaller players such as SCRUMStudy offer niche courses.

Organizations can turn to AgilityHealth, evidence-based management, Comparative Agility, Forrester, SAFe, the Agile Adoption Framework, and the Agile Journey Index (among others) to assess their level of agility.

Sales managers are agile. Training is agile [Gil13]. Librarians are agile [Mck09].

1.1 Crossing the chasm

As agile has spread, the backlash has been fierce.

A number of people have written about the ubiquity of agile and its subsequent loss of meaning. Dave Thomas, one of the original developers of the “Manifesto for Agile Software Development” or Agile Manifesto, has declared [Tho14], “Agile is dead.” Thomas suggests that agile “has been subverted to the point where it is effectively meaningless, and what passes for an agile community seems to be largely an arena for consultants and vendors to hawk services and products.” He suggests the word has been co-opted to boost sales in the same way that “green” has been used.

Stephen Cohen and Robert Galen have both asked if agile has jumped the shark [Coh11][Gal14]. Tim Ottinger has opined that he wants agile back [Ott14]. The Anti Agile Manifesto has been released as a parody site [Ant15]. Hayim Makabee wrote about the end of agile [Mak14].

A great rant from Tom Elders on *Hacker News* starts with [Eld12] “I can’t take this agile crap any longer. It’s lunacy. It has all the hallmarks of a religion.”

Andy Singleton at Assembla even wrote an article titled “Seven Things I Hate About Agile” to, in his words, “burn off the stink of stagnation” that surrounds the term [Sin12].

What is happening with agile?

According to the most recent “state of agile” survey from InfoQ, agile has gone mainstream and the majority of organizations use agile techniques for at least some software development projects.

We can use Geoffrey Moore’s chasm model for technology adoption to get a sense of what’s happened in the marketplace with agile. Moore’s model for disruptive technologies is useful because it looks at innovations that require people to do things differently — innovations that require behavior changes.

Looking at Moore’s model, innovators and early adopters are visionaries with a high willingness for change, high risk tolerance, and strong support from management. Early adopters understand the benefits and are willing to experiment in order to gain a competitive edge.

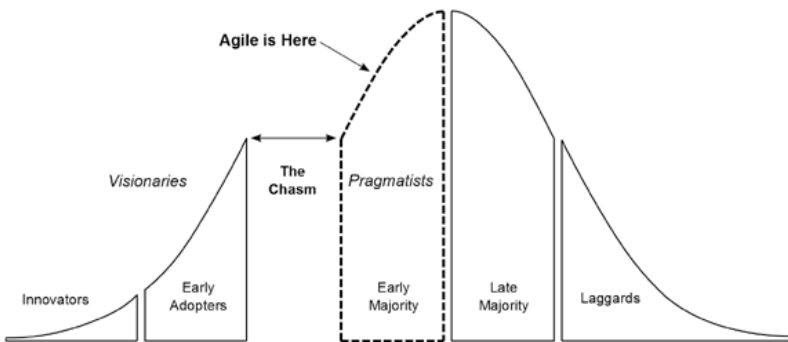


Figure 1.2: Geoffrey Moore’s model for crossing the chasm.

There is a large gap or chasm between these innovators and early adopters and the largest segments of the market: the early and late majorities.

Pragmatists and conservatives on the other side of the chasm are far more likely to approach agile from a completely different perspective. They are risk averse. They have heard of agile but likely think it is a process change that they can easily roll out to their IT organizations. Their risk tolerance is low, they want quick results, and they’re expecting relatively easy-to-implement process changes.

In other words, they are driven by practicality and want an out-of-the-box solution. The early majority wants technologies that are simple to implement.

As a result, many vendors and consultants have figured out that they can take advantage of the industry buzz and the early majority’s desire for practicality to sell agile tools and processes to convince these customers they are becoming more agile. As William Pietri wrote on the Agile Focus weblog [pie11], “An idea that provides strong benefits to early adopters

gets watered down to near uselessness by mainstream consumers and too-accommodating vendors.”

Much of this has happened in the agile marketplace as early adopters sought out-of-the-box tools and processes.

Coaches and consultants with experience in making the transition are spread thin and many new consulting organizations look to take advantage of the situation and sell their services.

The early majority also sees agile as a process to enhance productivity rather than a potentially disruptive culture change. Agile can (depending on existing culture) be a significant cultural change. Crossing the chasm is more difficult with agile than with other innovative technologies because organizations might not have a culture that is ready for agile and either don't understand or underestimate the cultural change inherent in agile.

The Agile Manifesto reads:

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

The Agile Manifesto describes a change in beliefs, a cultural change.

Tobias Mayer described it this way in *The People's Scrum*:

Scrum is a framework for organizational change and personal freedom. It is not a methodology, it is not a process, and it is much more than a tool.

Agile is a set of beliefs, a set of ideas. Are executives and leaders willing to adopt and champion these ideas? Or are they merely looking to “optimize” employees because employees are seen as the constraining element of the system?

If you look at agile-consulting organizations, how many of them are process, tool, or methodology heavy? How many of them want to sell a system for doing agile?

As Dave Thomas writes [Tho14]:

Now look at the consultants and vendors who say they'll get you started with "Agile." Ask yourself where they are positioned on the left-right axis. My guess is that you'll find them process and tool heavy, with many suggested work products (consultant speak for documents to keep managers happy) and considerably more planning than the contents of a whiteboard and some sticky notes.

Moore's ideas about crossing the chasm help us understand that what is happening is normal for innovations that impact behavior.

We don't believe agile is dying or jumping the shark, but rather is experiencing growing pains as it reaches new markets. In many cases, however, what this means to organizations on the other side of the chasm is that what they're doing or attempting to do is not really agile.

1.2 Adoption vs. transformation

One of the more common mistakes made when implementing agile is not seeing it as a framework for organizational change. This typically looks like adopting sprints and the artifacts associated with sprints and ignoring other components of the change framework, most often agile values.

When asked why agile projects fail, the number two reason cited in VersionOne's 2014 "State of Agile Survey" after "None of our projects failed" was "Company philosophy or culture at odds with core agile values."

Henrik Kniberg tells the story of one of his most successful projects — a system built for the Swedish police that allowed them to use laptops in the field — and what happened afterwards [Kni13]. Because the project was extremely urgent, the group was allowed to use an agile approach and break out of the traditional organizational culture. Everything went well, the police organization viewed it as a success, and the project even won a "project of the year" award.

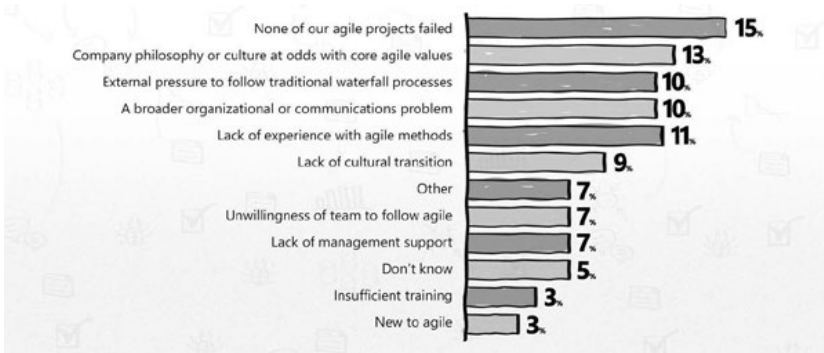


Figure 1.3: From VersionOne's 2014 "State of Agile Survey".

What came next, however, was even more interesting. A high-level decision was made to rebuild from scratch that same system police had used in the field, using Siebel. This was part of a standardization effort to reduce the complexity and number of systems. Not only was the decision made to use a technology that the development team didn't agree with, but it was decided to use a more traditional, sequential project-management approach to development. Development took a couple years and when it finally rolled out, it was a disaster because the police found it to be slow and clumsy and basically unusable. Making the change even more difficult was that the police preferred their existing system, which worked. Kniberg estimates that this cost the Swedish police more than £1 billion.

Adopting agile practices is likely to lead to marginal improvements at best if current values and culture are out of alignment with agile beliefs and the organization doesn't change.

Similarly, when asked about barriers to further adoption, inability to change organizational values was cited as the top barrier in VersionOne's 2014 survey.

As Mike Cottmeyer wrote in "Untangling Adoption and Transformation" [Cot11]:

- Transformation is about changing the "agile being" side of the equation.
- Adoption is about changing the "agile doing" side of the equation.

Some symptoms that might indicate that transformation has not yet fully happened and agile culture and values have not yet been adopted are:

- Agile teams have defined dates and scopes.
- A manager assigns tasks to team members.
- Impediments to development are not addressed.
- Team members don't point out problems when they see them.
- Testing is not allowed because it highlights shortcomings.
- Burn-down charts are altered to present a rosy picture.
- Management plans rather than teams.
- All features are seen as high priority.
- Communication is one way, from leaders to employees through broadcasts.
- Agile is seen as something “the technology people do”.
- Teams are not developing working software.
- Teams are reporting rather than discussing progress.
- Superstars are valued over team.
- No changes affect how things are done.
- There is a reluctance to hire qualified outside experts.
- Leadership demands results without providing direction.
- Knowledge is hoarded.

To realize the full benefits of agile requires the values or the “being” part of agile.

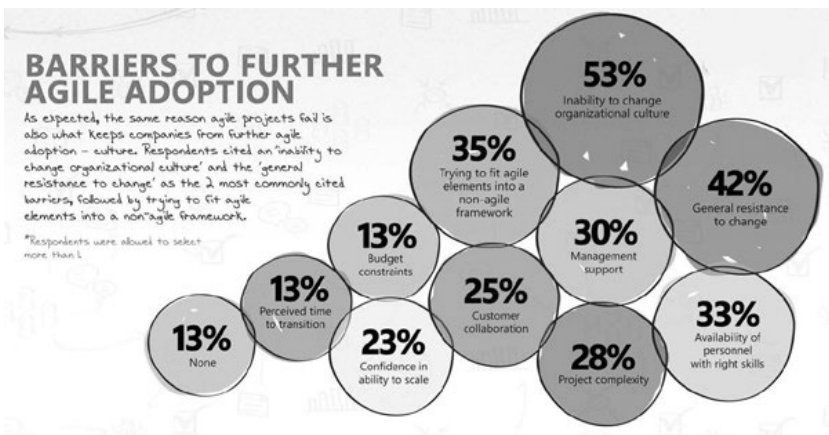


Figure 1.4 Barriers to agile adoption, from VersionOne’s 2014 “State of Agile Survey”.

Michael Sahota and others have discussed how agile processes and methods can be adapted to different cultures [Sah04]. We would like to take a

different approach. We believe that if organizations adopt agile as a set of beliefs, they will develop an agile culture and that this agile culture is what leads to continuous adaptation and innovation. The focus of the change effort must be on the heart, not the head or the hands.

Processes and methods can become stale and rote, and can stifle innovation — even processes that were initially developed to be agile. An agile culture, however, will continuously improve and adapt without the need for periodic change initiatives.

Numerous books and best practices exist to help organizations with implementing agile practices, or the “doing” side of the equation. Our reason for writing this book is to examine the values and culture that make organizations agile.

1.3 How the rest of the book is organized

Chapter 2, “Agile Values”, describes the difference between values and beliefs, discusses why organizations should care about culture, outlines the results pyramid and Fredric Laloux’s model of organizational values, and introduces a simple framework for agile values.

Chapters 3 to 6 discuss the four major components of the agile values framework (trust, responsibility, learning, and collaboration), use examples to show what these look like in agile organizations, and highlight differences in how they are often interpreted.

Chapter 7, “Agile Values Revisited”, summarizes key learnings and recommendations for organizational change.

“Additional Resources” contains other materials that can help with agile transformation and key charts and questions to ask about organizational values.

Chapters 3 to 6 can be read in any order.

PART TWO

Agile Values

“I believe that we all have the potential to solve problems and express ourselves creatively. What stands in our way are these hidden barriers — the misconceptions and assumptions that impede us without our knowing it.”

— Ed Catmull, *Creativity, Inc.* [Mic15]

2.1 Values and beliefs

To explore organizational culture and how it influences performance, we’d like to start by looking at individual values and beliefs — not specific values or beliefs, but rather what these terms mean.

People associate values with character. We associate values with ethics. We associate values with who we are. Someone who has values is someone to look up to, someone that is like you or that you want to be like, someone who lives his or her life in a certain way.

In *On Value and Values*, Douglas K. Smith writes [Smi04]:

Values are nouns, but nouns concerned with verbs of attitude and action. Values sort into several categories. People refer to social values and political values; and, to family and religious, and environmental values. Values are estimations not of worth but of worthwhileness. Unlike value, talk of values ignores money; it opines on timeless appraisals instead of transient ones. There is a deep backward- and forward-looking quality to values. If value is what makes us wealthy, values, we assume and regularly assert, are what makes us human.

Examples of values include:

- honesty,
- loyalty,
- learning,
- trust,
- family, and
- leadership.

A belief, by way of comparison, is something we hold to be true.

Both values and beliefs guide our actions and behavior. Because they seem so similar, the difference between the two can be confusing.

Beliefs come from our experiences with our families, our culture, our communities, our education, and our jobs. Examples of beliefs include:

- If you want people to treat you well, treat them well.
- Knowledge is power.
- The world is made up of idiots and we're two of them.
- People make decisions based on facts.

If beliefs are things we hold true, values are what we believe are important. The connection between the two is that beliefs influence values and how we prioritize values.

For example, someone who believes that knowledge is the key to a better life is probably going to place a high value on education. If we believe predicting the future is impossible, we tend to value experimentation over planning.

Similarly, we can think of organizations as having values and beliefs. Instead of individual values and beliefs, which differ from person to person, these are the shared values and beliefs that determine how an organization performs.

2.2 Values unite, beliefs divide

An important distinction between values and beliefs is that values unite people while beliefs tend to divide people. This is because people tend to have a similar set of values even though they may be prioritized differently. For example, it's hard to argue that collaboration is important. However, if you made a true/false statement of belief like "Collaboration is more important than safety," some people would agree and some disagree depending on their own experiences. Asserting something is true or most important can be a point of contention because people tend to feel passionate about what they've learned through their experiences.

Think about how hard it is to get a group of people to unite around different religious beliefs or different political beliefs or different cultural beliefs. They will fight morning, noon, and night defending their beliefs.

Yet if you ask them how they feel about “family” or “freedom” — values that everyone holds in some way, shape, or form — you tend to find wide support. Thomas Jefferson, for example, united people by finding a shared value of people of faith: freedom of religion. Jefferson wrote the Virginia Statute for Religious Freedom [Act86]:

Be it enacted by the General Assembly, that no man shall be compelled to frequent or support any religious worship, place, or ministry whatsoever, nor shall be enforced, restrained, molested, or burthened in his body or goods, nor shall otherwise suffer on account of his religious opinions or belief; but that all men shall be free to profess, and by argument to maintain, their opinion in matters of religion, and that the same shall in no wise diminish, enlarge, or affect their civil capacities.

Jefferson’s statute became the basis for the First Amendment of the U.S. Constitution. This work was one of only three accomplishments he instructed be put in his epitaph.

Here’s an exercise you can conduct to demonstrate the difference between values and beliefs. This exercise can be done with two or more people. You will need Post-it notes for everyone involved. To practice, you might want to first try it with one other person.

1. Take five minutes and write down as many of your values as you can think of, one to a Post-It note. Don’t worry if you can’t think of everything. This is not a competitive exercise. It’s a collaborative exercise. If at any point during the collaboration, you think of something you forgot to write down, you can add it.
2. Look at what you’ve both written down and combine all of the values that are the same.
3. Look at what you’ve each written down that might be different, but that you agree is a value you also hold. Add all of these Post-its to the combined list of values. These are the values you share. Don’t dwell on any Post-its that are not in common.
4. Things to think about:
5. How similar or different are your values?
6. Were any of the values listed statements of truth or beliefs?

Figure 2.1 is an example of values shared by two people who have very different backgrounds.



Figure 2.1: Combined values exercise.

In this example, duplicates included:

- family
- friends/friendship
- relationships/trust/“I’ve got your back” kinds of relationships
- love
- helping others/activism/commitment to the good of others
- “a higher calling”/purpose
- equality/all “men” created equal

Other values that both participants agreed upon included:

- generosity
- kindness
- honesty
- faith
- peace
- fun

- faithfulness
- freedom
- responsibility
- good health
- work
- life
- government “of the people, for the people, and by the people”
- education
- leadership/strength

The result of this exercise is a powerful visualization of shared values. While the participants might have disagreed on specific beliefs due to their different backgrounds and experiences, their overall values were very similar.

In Jefferson’s day, to get different religions that believed different things to unite as a country, it was necessary to find values that they all held in common. One such value was freedom of religion: everyone should be free to practice their religion so long as beliefs didn’t interfere with other people’s freedom.

This distinction between values and beliefs is important to highlight as we begin to consider organizations because individuals will have different values and beliefs and these values and beliefs may be quite different from an organization’s. This difference is perfectly normal. However, it’s important to note because, more often than not, it’s discussions about beliefs that cause friction. If you want to find points of commonality, it is easier to start from values than beliefs.

2.3 Organizational culture determines results

In *Change the Culture, Change the Game*, Roger Connors and Tom Smith describe organizational culture as an organization’s experiences and beliefs.

They then visualize the relationship between beliefs and results using a results pyramid. In this model, experiences foster beliefs, beliefs influence actions, and actions produce results.

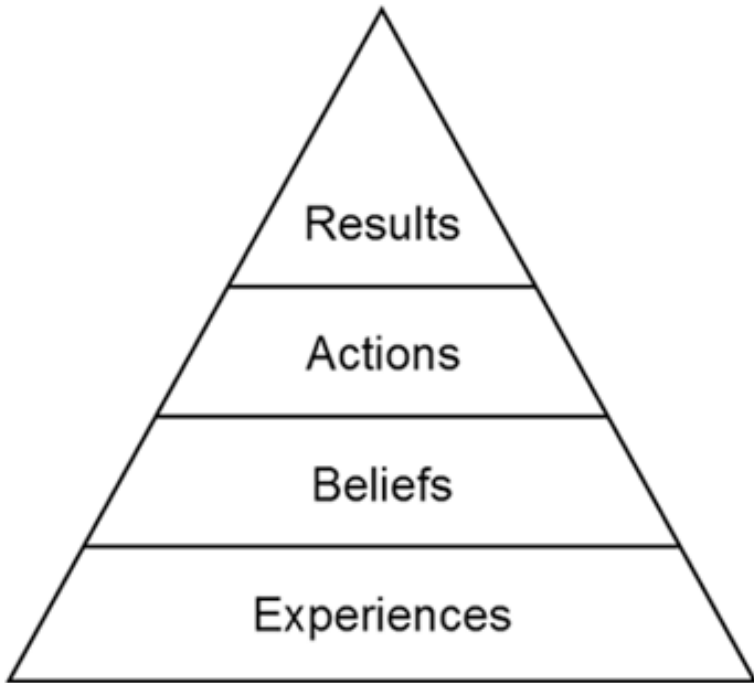


Figure 2.2: Roger Connors and Tom Smith's results pyramid.

In this model, it's the culture (experiences and beliefs) that really produces results. They describe it this way:

Culture depends on results; results depend on culture. Leaders can build a company culture around any set of desired results: market dominance, sales growth, technological excellence, ease of customer interaction, best-in-class quality, or stable earnings, just to name a few. Once you clearly define the targets, then you must move quickly to build a culture that produces the right experiences, beliefs, and actions to achieve those results.

The authors used a pyramid instead of a simple hierarchy to highlight the fact that culture plays a much bigger role in determining results than actions do because actions depend on culture. Culture determines how and what employees do in a given situation.

A classic mistake that organizations often make when trying to improve performance is to focus only on the top of the pyramid. Organizations often change processes (actions) while ignoring the fact there are reasons why people think and act the way they do. This disconnect can cause significant issues if the processes are out of alignment with the core culture.

If the organization only focuses on actions and new actions go against organizational culture, process changes are unlikely to last or have meaningful impact.

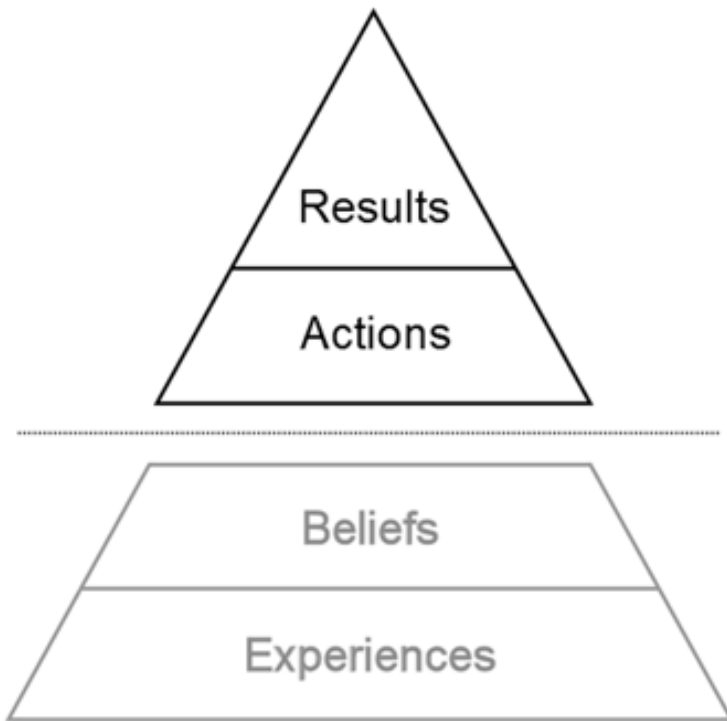


Figure 2.3: Often, organizations focus on the top.

A great example is a hospital that wanted to improve the speed of its surgical-tray sterilization procedures. Without talking to the people who sterilized the instruments, management ordered several million-dollar robotic tray systems. Had they talked to the people in the sterilization department, they would have learned that the real problem was declining morale because the staff viewed new managers as micromangers. Instead of solving the problem, the decision only reinforced the organizational view that employees were not to be trusted. The new robotic

tray dispensers also greatly slowed the process because each could only dispense one tray at a time; previously, multiple employees could retrieve surgical instrument trays.

The hospital could have avoided millions of dollars in process and technology changes and increased speed had management been able to recognize the cultural issues and simply asked and involved employees. The micromanagement culture led to millions of dollars of waste.

To truly transform an organization requires working with the pyramid's full depth and breadth.

In *The Culture Game*, Dan Mezick expands beliefs into beliefs, values, and principles. In Mezick's version of the results pyramid, similar to individual values and beliefs discussed above, organizational beliefs inform values and their prioritization. Beliefs inform organizational values and people develop principles or heuristics based on these values.

Experience, beliefs, values, and principles form the organizational culture that determines actions and results.

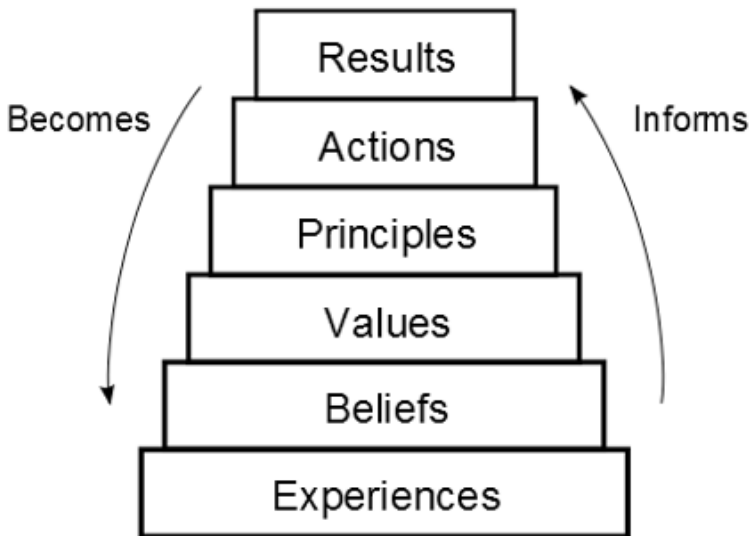


Figure 2.4: Mezick's version adds values/principles and the feedback relationship.

Mezick also depicts the feedback loop of the relationships. Results are not simply a one-way relationship but rather a feedback loop that is constantly building as results become organizational experience.

An organization that tries something new and gets good results might revisit some of its beliefs so that these new beliefs become part of the culture. This process is organizational learning.

Adding values and principles is critical to highlight because as beliefs change how we prioritize, our values change. As beliefs change, priorities change. This, in turn, influences principles (or rules of thumb) and guides our actions and results.

Expanding beliefs into beliefs, values, and principles tells us more about organizational culture and how it works and evolves.

Often, some of the organization's values and beliefs are written down in a mission or values statement. A few examples include:

- “Create fun and a little weirdness” and “Deliver WOW Through Service” — Zappos [Zap15].
- “Trust and personal responsibility in all relationships” — IBM [IBM15].
- “Build the best product, cause no unnecessary harm, use business to inspire, and implement solutions to the environmental crisis” — Patagonia [Pat15].
- “Don't be evil” — Google's former slogan, which they dropped in 2009 [For09].
- “Inspiring humanity” —jetBlue [Jet15].
- “Offer the customer the best possible service, selection, quality, and value” — Nordstrom [Nor15].

It's just as easy, however, to find mission statements that don't represent the actual organizational culture. One of the more famous recent examples is the culture at British Petroleum and the Deepwater Horizon oil spill.

In 2009, part of BP's mission statement read [Ama13] “We aim for no accidents, no harm to people, and no harm to the environment.”

From the *Final Report on the Investigation of the Macondo Well Blowout* [Dhs11]:

Analysis of the available evidence indicates that when given the opportunity to save time and money — and make money — tradeoffs were made for the certain thing — production — because there were perceived to be no downsides associated with the uncertain

thing — failure caused by the lack of sufficient protection. Thus, as a result of a cascade of deeply flawed failure and signal analysis, decision-making, communication, and organizational-managerial processes, safety was compromised to the point that the blowout occurred with catastrophic effects.

The report cited BP's corporate culture as the reason for the Deepwater Horizon oil spill. They claimed to have a culture of safety. In reality, however, the culture was about deadlines and cutting corners to make money.

More often, the difference between written culture and the unwritten culture isn't quite as extreme. One example is a company that promotes teamwork while evaluating and rewarding based on individual performance. Another common example is companies where employees work 12-hour to 16-hour days while the company promotes itself as having a healthy work/life balance.

Organizational culture is the set of shared experiences, beliefs, values, and principles — both stated and unstated [Dun14] — that determines results.

2.4 Agile is a set of beliefs

Agile is a set of beliefs and principles, a set of statements that a group of people proposed to be true from their experience about complex software development.

Again, the Agile Manifesto states [Fow01]:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

These statements are broad belief statements about what should be valued. These four statements compare two organizational worlds. A world that values individuals over processes and tools is going to look very dif-

ferent from a world that values processes and tools over individuals. The latter world views organizations as machines and the ultimate goal is to find the the perfect set of processes to make the machine run smoothly. This world believes that an organization is like a machine and people within the organization are like cogs or parts that can be optimized.

Valuing individuals and interactions defines a very different organizational world. In this world, an organization is more like a family. When we think of families, we don't typically think about defining processes that everyone in the family needs to follow. We don't think about maximizing productivity by implementing new management routines for each family member. We tend to think more in terms of how each member of the family can realize full potential.

Just this one statement about individuals and interactions shifts how we think.

While most people try to understand the processes in their approach to agile, we would like to take a different approach and outline what this world looks like from a cultural standpoint.

The four main Agile Manifesto beliefs are high-level beliefs that focus on prioritizing values. The Manifesto also includes 12 agile principles, heuristics, or best practices about how to best develop working software:

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity — the art of maximizing the amount of work not done — is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

These principles are best practices for development compared with the four higher-level value-ordering beliefs.

As Dan Mezick described in his results pyramid, principles are more like heuristics that, in this case, guide software development, while beliefs are higher-level statements of truth. For example, the fifth principle elaborates on the meaning of valuing individuals and interactions over processes and tools. It provides specifics on how to successfully complete projects: give people the environment and support they need and trust them to get the job done.

Principles follow from values follow from beliefs.

If beliefs influence values and values inform principles, as we've seen, one of the natural questions that arises is what do the organizational values look like in an agile culture? How does adopting agile beliefs affect an organization's values? What does a culture that values individuals and interactions over process and tools look like?

If we view agile as a philosophy or a set of beliefs, implementing agile might well lead to organizational change beyond adopting a set of processes or methods. It might well lead to a desirable change in organizational culture and values.

When groups adopt agile beliefs, the adoption process might change or influence the culture depending on the organization's existing culture. Or the culture may influence agile adoption. As we've seen, organizational culture is the primary reason cited for failures in agile transformation.

This point is also critical to understand because many people still see agile differently. They see it as a process or methodology. They see agile as something that affects actions, not necessarily as a framework for change. They don't see it as a set of beliefs with the potential to change their culture.

If an organization is approaching agile from a process or methodology perspective, it is not likely to get the desired results. Similarly, if the existing values in an organization are far from alignment with agile values, the effort is likely to either fail or be significantly greater than anticipated.

2.5 The evolution of organizational culture

Fredric Laloux, in his book *Reinventing Organizations*, outlines a history of organizational cultures [Lal14] as part of his goal to create better organizations. Through his research and that of others, he discovered that organizations tend to evolve in stages. All models and research strongly converge on and support this stages theory.

Laloux describes the stages with colors ranging from magenta to teal with everything before magenta represented by the invisible-to-the-eye infrared, a stage where the only communal relations were basically familial.

Figure 2.5 illustrates this evolution of organizations along a historical timeline.

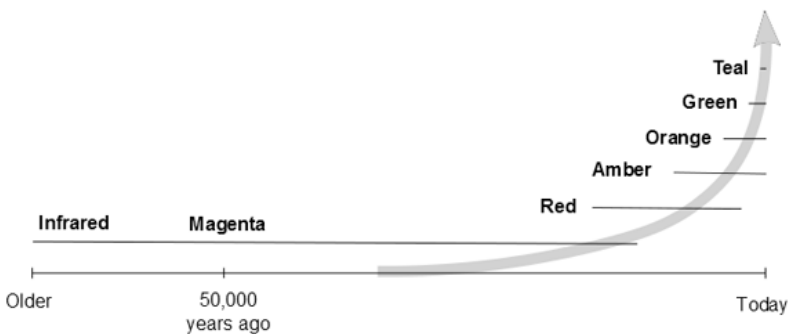


Figure 2.5: Laloux's evolution of organizations.

Laloux’s model includes the challenges that confront organizations and the innovations that lead to subsequent breakthroughs towards the next stage. In addition, he describes the characteristics of the culture and many of their values.

The model provides a way to look at organizational development by looking at the history of how different types of organizations evolved, the challenges they faced, and how they innovated to adapt to these challenges. The model also provides a way to look at some of the different value priorities in each of the different stages.

Table 2.1 lists the characteristics and breakthroughs of each stage.

Organization	Breakthroughs	Characteristics
Red (wolf pack, mafia)	<ul style="list-style-type: none"> • Division of labor • Command authority 	Strong, tough, dangerous, power, armies, street gangs, mafia, force
Amber (army)	<ul style="list-style-type: none"> • Long-term perspective and processes • Size and stability (formal hierarchies) 	Static, fear, right or wrong, institutions, bureaucracies, castes, social classes, order, predictability, roles, guilt, compliance, army, hierarchy, processes, certainty
Orange (machine)	<ul style="list-style-type: none"> • Innovation • Accountability • Meritocracy 	Achievement, empirical, science, expertise, truth, entrepreneurship, modern, innovation, materialistic, accountability, meritocracy, rationality, performance
Green (family)	<ul style="list-style-type: none"> • Values-driven culture and inspirational purpose • Empowerment and multiple stakeholder • Perspective 	Feelings, fairness, equality, postmodern, harmony, community, cooperation, consensus, relationships, service, values-driven, purpose

Teal (living system)	<ul style="list-style-type: none"> • Self-organizing • All breakthroughs of previous stages 	Trust, collaboration, service, pride, networks, learning, self-organizing, wholeness, community, fun, purpose, power, teams, facilitators/coaches, ownership, responsibility, resilience, compassion
-------------------------	---	--

Table 2.1: Characteristics of organizations (red to teal).

Primitive groups in the infrared stage were largely familial and subsisted largely on foraging. At this level, there is little division of labor and therefore almost no organizational model. The concept of self is virtually indistinguishable from others.

The magenta level is the tribal level. At this stage, groups shift from small family units to groups of up to 100 people. Authority figures, such as elders and shaman, arise but there is little organization beyond that.

Most of the organizations we’re familiar with are red through green organizations.

In developed societies, red organizations typically exist on the fringes of legality. They look like street gangs or mafias — non-trusting, tough, dangerous, and forceful. Amber organizations are still commonly found among government agencies, the military, public schools systems, and religious organizations. Orange organizations are dominant in the corporate and business world. Green organizations are common in non-profits and have also been making inroads in the business world. Ben & Jerry’s and Patagonia are a couple examples of businesses that fit well within green paradigm.

Teal organizations, which Laloux sees as emerging, are still uncommon.

In terms of Laloux’s model, agile requires a green or teal organizational culture for transformation. It could also be said that agile beliefs define a teal culture that, if not realized, undermines successful transformation. Organizations that have the most success with agile are either at the green/teal stages or are moving towards the green/teal stages.

From our experience, what Laloux describes as a teal culture and what we’ve seen of agile cultures are very similar.

Our goal is to specifically bring to the surface and outline with examples the values of agile organizations to make them clear and easier to understand. For instance, collaboration means something very different in a teal organization than in an orange organization.

Describing the values of agile organizations not only helps transitional organizations understand whether they are moving in the correct direction but it also helps organizations spot cultural conflicts that might prevent agility.

2.6 A quick digression

Before discussing values of an agile culture, we'd like to address a question that tends to come up in any discussion of values.

Bluntly, the question is "Is one set of values better than another?"

No. Laloux does not advocate one organizational model over another, but rather he looks at how and when organizational paradigms map well to different situations. Similarly, we see our efforts as a way to look at agile organizations and talk about how these organizations define themselves through their values and how these values influence decisions.

Second, we'd like to make a clear distinction between organizational values and individual values. As Laloux states [Lal14]:

I'm referring to systems and culture, not people. If we look at an organization's structures, its practices, its cultural elements, we can generally discern what worldview they stem from.

At any moment in time, different people act on different values and principles depending on their situation. People tend to switch back and forth quite easily between norms depending on the group they are interacting with.

Here, we are talking about the values of organizations and, in particular, what organizational values are required for agile transformation.

2.7 A values framework for agile transformation

Agile is more than a tool to achieve better results. Agile is a framework for organizational change that leads to a more human-centered organization.

We have discussed beliefs and values, and how agile is a set of beliefs. But what do these beliefs lead to? What does an agile organizational culture look like?

To come up with a simple framework for agile values, we drew on our own experiences with teal, green, orange, and amber organizations and a set of criteria for considering values and their relationship to the Agile Manifesto.

The criteria for our framework is as follows:

- Are the values consistent with our experience and the experience of others in agile organizations?
- Do the organizational values follow from Agile Manifesto beliefs or do they support any of the Agile Manifesto's 12 principles?
- Is our value list simple and useful?
- For our purposes, simple means simple enough to be easily committed to memory so that the list is easy to mindfully practice with the ultimate goal of guiding behavior, decisions, and actions.
- Completeness may be sacrificed for the sake of simplicity if simple covers almost or near enough values.
- Useful means does it help people understand what the core organizational culture should look like to support agile transformation?
- If we removed a value from the organizational culture, would it be significantly less useful? In other words, would it seriously impact agile transformation?

Using these criteria, we came up with the four values in Figure 2.6.

Aspects of these values are similar to the five values the Scrum Alliance defines: focus, courage, openness, commitment, and respect.

When we looked at existing models of agile organizational values and applied the criteria above, we found current models incomplete. Many contained pieces or parts of the whole, but nothing presented a full picture.

For example, the Scrum Alliance values don't specifically mention learning as an organizational value. Jeff Sutherland and others, however, have discussed the importance of rapid iterative learning and retrospectives [Sut14].

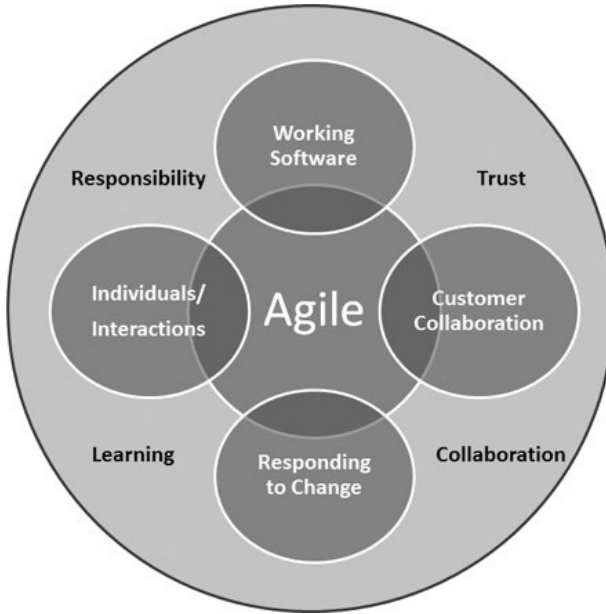


Figure 2.6: Agile values framework.

Each of these four values — responsibility, trust, learning, and collaboration — has several aspects that are critical for agile transformation. A short list of these is shown in Table 2.2.

Trust	Responsibility	Learning	Collaboration
Openness	Autonomy/ Freedom	Risk	Transparency
Credibility/ Integrity	Motivation	Feedback	Self-organization
Craftsmanship	Commitment	Adaptability	Communication
Empathy/ Respect	Mutual Responsibility	Sharing	Unity/ Shared Purpose

Table 2.2: Aspects of agile values.

We found it helpful to pull together a single broader picture.

Throughout this chapter, we've highlighted reasons for putting together this cultural model of agile values. Here, it helps to summarize the reasons:

1. Agile is crossing the chasm and experiencing growing pains with more conservative organizations.
2. Many people and many of these organizations see agile as a process or a methodology.
3. Organizations should understand that agile is a framework for change when they are deciding whether to pursue transformation.
4. Agile adoption (doing) is very different from agile transformation (being).
5. Cultural conflicts are cited as the number-one reason agile projects fail and inability to change organizational culture is the top barrier to agile adoption.
6. Culture determines results.
7. An understanding of agile culture can help executives make better decisions about agile.
8. Much literature has been written about implementing Scrum and XP and kanban and relatively little has been written about agile culture.
9. A better understanding of the destination might improve the journey.

In subsequent chapters, we'll dive deeper into the model and share examples where appropriate to paint a picture of agile culture and talk about some of the differences among amber, orange, and green cultures. The goal is not to provide a comprehensive picture but to provide a useful picture.

What do these values look like, what are some of the beliefs that might undermine them, and what beliefs can we introduce and model to lead to successful agile transformations?

PART THREE

Trust

“The best way to find out if you can trust someone is to trust them.”

— Ernest Hemingway

3.1 Theories X and Y

An interesting question to ask people is whether they believe people 1) are lazy and tend to avoid work, or 2) are ambitious and self-motivated.

It’s a great question because it makes people think. When you ask it, most people tend to lean one way or the other and after a minute or so of thinking about it say, “It depends.”

“It depends” usually means it depends on the person.

Douglas McGregor, in his book *The Human Side of Enterprise*, proposed that a manager’s assumptions or beliefs about human nature determined a management style [Mcg60].

McGregor developed two theories of management, Theory X management and Theory Y management. Theory X leaders assume that people:

- dislike and will try to avoid work whenever possible;
- are lazy and prefer to be directed;
- are not creative or natural problem solvers; and
- must be coerced, controlled, directed, or threatened with punishment to get them to put forth sufficient effort.

Theory X leaders tend to take either a hard or soft approach to motivation. The hard approach is command and control: coercion, implicit threats, micromanagement, and tight controls. In two organizations that have been models for command and control, the army and the church, authority has been enforced through punishment such as court martial in the army or excommunication in the church.

The soft approach is to motivate through money in exchange for cooperation. The hard approach results in hostility and purposely low performance. The soft approach results in increasing demand for rewards in exchange for diminishing work output.

Abraham Maslow's hierarchy of needs is a way to think about individual growth as progressing upward as different needs are met or satisfied [Mas43]. At the bottom of the pyramid are physical needs, such as food, sleep, and water, and safety needs, such as protection and health.

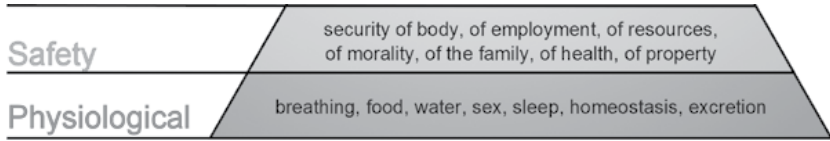


Figure 3.1: Maslow's physiological and safety needs.

Both Maslow and McGregor argue that when a need is satisfied, motivation lags or disappears. In modern society, needs at the bottom of Maslow's hierarchy are in many ways satisfied and might no longer provide sufficient motivation. When dependence is complete, as in a parent/child relationship, the "because I said so" model will work for a while. As people progress upwards in Maslow's hierarchy, however, authority no longer has the same effectiveness and reliance on authority might encourage countermeasures and lower performance.

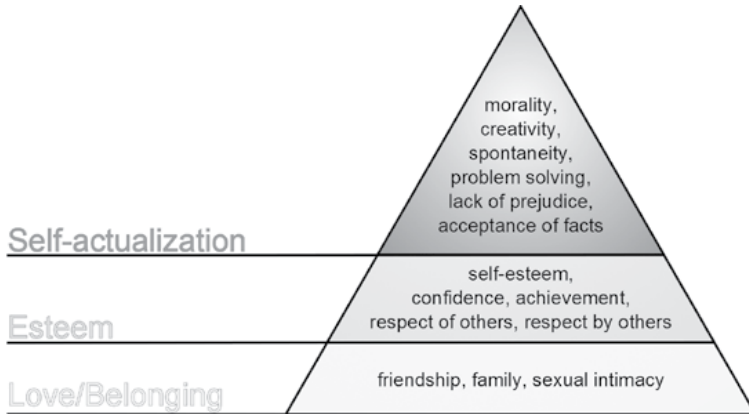


Figure 3.2: The top of Maslow's hierarchy of needs.

Theory Y management is based on the top of Maslow's hierarchy of needs and focuses more on what individuals want to achieve for themselves.

Theory Y leaders believe:

- Work can be fun and a source of satisfaction.
- People are motivated and self-directed in the service of objectives to which they are committed.

- People learn not only to accept but also to seek responsibility.
- People are creative and natural problem solvers.
- Motivation occurs by looking for ways to match an individual's personal needs with organizational needs.

McGregor recognized that some people have not reached the level of maturity that Theory Y assumes, and might need tighter controls until individuals develop.

Theory Y suggests that organizations will be more effective if they acknowledge and accommodate their employees' needs and goals. Modern organizations can do this with flexible hours, retirement programs, health care, onsite fitness centers, dedicated time and budget for training and self-improvement, and maternity leave. Theory Y managers tend to be more hands off. They allow employees a great deal of freedom to achieve results. These managers recognize individual needs for autonomy and let employees manage themselves. Theory Y managers often involve employees in decision making as well.

One point to note here is that results tend to reinforce assumptions. That is, if a manager holds Theory Y's assumptions and establishes working relationships within which individuals are trusted and encouraged to do what they do best, people will be more motivated and self-directed. If managers hold Theory X's assumptions and micromanage people, the outcome will tend to confirm the manager's suspicion that people are lazy.

The reality is often more complex than this, but this feedback loop provides a valuable way to look at assumptions and their impact on organizations.

At the heart of McGregor's Theory Y is trust in individuals.

3.2 Trust equals speed

People often think about trust as a "touchy feely" to have far down the priority list in our age of business, hard data, numbers, and reasoning.

Stephen M.R. Covey, son of Stephen R. Covey of *The Seven Habits of Highly Effective People* fame, uses these two simple formulas to make the business case for prioritizing trust [Cov06]:

- $\downarrow \text{Trust} = \downarrow \text{Speed} + \uparrow \text{Cost}$

- $\uparrow \text{Trust} = \uparrow \text{Speed} + \downarrow \text{Cost}$

As trust increases, speed increases and cost decreases.

A great example of these simple formulas is the regulations passed in response to the Enron, Tyco International, and WorldCom scandals. These companies and others were cooking the books and when investors found out, it led to a collapse in their share prices, which cost investors billions and shook confidence in the entire U.S. stock market.

The subsequent laws passed restored trust in the system but this trust came at a cost. The cost is compliance and record keeping associated with Sarbanes-Oxley. Our point is not to argue in favor or against the act but merely to demonstrate that if there were a way to increase trust between companies and the public without this act, speed would go increase and cost would drop.

You can think about this in your own relationships at work.

- How much faster is it to get help from colleagues if you know them and have a trusted relationship?
- How much easier is it to ask people you know well for help?
- When you don't know someone or have a trusted relationship, what do you have to rely on?

Often working with people who are located far away, we sometimes use virtual teams. The most critical thing we learned with virtual teams was that we need a significant amount of time (at least a week) to kick off a project in person.

At kickoffs, it is important to lay out goals, prioritize actions, and figure out an initial commitment plan. The most important thing, however, is to build working relationships with the people on the virtual project team. We learned we could get away with not planning everything. Project plans can always be adjusted if you work with people you trust.

Failing to establish trusted relationships, however, is a completely different animal. Failing to develop trust early will almost always lead to complications. This point might seem obvious, but think about how much time people typically spend on planning and how much they spend on building relationships at the beginning of projects.

For reasons like these, agile specifically values individuals and interactions over processes and tools. Trust is also established through these agile principles:

- Face-to-face communication — the most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Trust between business people and developers who “must work together daily throughout the project”.

These principles help build relationships and levels of trust. With a virtual team, trust is even more important. Why? Because it’s harder to have the higher-touch interactions that agile specifies as critical.

Does this mean it’s not possible to follow agile with virtual teams? No. What it takes, however, is a significant level of trust and strong relationships. This reality is why we always prioritized at least a week for face-to-face meetings and relationship building with virtual team projects.

One of the reasons virtual teams are often discouraged in agile is because it is so much easier to build trust and strong relationships in a face-to-face setting. When teams work in proximity and trust each other, they have an almost instant understanding of where everyone is and what everyone is doing and they can immediately deal with any issues that arise.

These agile principles help build relationships and levels of trust.

3.3 Aspects of trust

Briefly, Table 3.1 illustrates some high-level aspects and signs of trust in organizations and their importance to agile.

Aspects of Trust	Importance to Agile
Openness	Openness is necessary for continuous improvement. If something isn’t working or could be improved, maybe there’s a better way. If people can communicate the difficult, issues can be addressed.
Credibility/ Integrity	Credibility enables business people and developers to work together daily. Integrity leads to credibility.

Craftsmanship	Craftsmanship is one of the first ways people involved in a shared project learn to trust each other, through delivery in small increments.
Empathy/Respect	Respect and empathy allow people to understand each other. This is part of valuing individuals and interactions over processes and tools.

Table 3.1: Aspects of trust.

The next four sections look more closely at these aspects of trust.

3.3.1 Openness

Trust increases when motives are straightforward and based on mutual benefit.

- Do you hear the bad as well as the good?
- Are issues raised or hidden?
- Do people understand why things are done? How transparent are decisions?

In an agile culture, it's important to be honest in order to delight customers. If issues are raised early, you can address them. Also, sometimes what you hear are symptoms rather than actual issues. It might take a little digging to get to the issue and this occurs much more quickly if you can have honest conversations.

Michael on the difference between openness and honesty

To me, there is a difference between openness and honesty. The difference is similar to the difference between answering a question and answering the intent behind a question.

Here's a short example to clarify. I was the lead coach in an engagement and was responsible for bringing in other agile coaches. During the process, coaches wanted to know how long the engagement would last. Obviously, this information would help them plan their lives.

I was working for a consulting company, whose official answer was, "The client would like you to be here for a year."

Now, I had data on coaches who had worked with this particular client, which showed that 70% of coaches left within six months for a variety of reasons.

An honest answer to the question would be to simply repeat the official line from the consulting company about how the client was looking for coaches for a year. An open answer to the question, however, is to say, “The client would like you to be here for at least a year. However, 70% of coaches leave within six months.”

The open answer might lead to follow-up questions such as “Why do coaches typically leave within six months?” Typically, people want to know the length of an engagement because they need to make calculations about their lives. Providing them with the facts helps them decide.

Open answers often lead to difficult conversations and this is a good thing.

David on how openness leads to addressing issues

As an instructional designer, I was once asked to improve the strategic quarterly review meeting for the learning organization. The executives wanted to improve the meetings and make them more engaging so they asked our group if we could use technology to achieve that.

My director called me and said, “We’ve been asked to use technology to improve the quarterly review meetings.”

I laughed.

“Why are you laughing?” my director asked.

“Because I don’t think any amount of technology is going to fix that meeting,” I said.

Having been to past meetings, I knew the issue wasn’t a lack of technology. The issue was that the meeting took an entire day and the format was boring. Each presenter was given an hour and each felt obligated to use the entire hour so it was a series of roughly eight hour-long presentations. No amount of technology was going to fix this problem.

After a quick discussion, my director agreed with me. This open conversation, in turn, allowed me to ask, “Is technology really the right solution for making the meeting better?” We agreed that it wasn’t. The meetings needed to be shorter and more focused.

We decided to run the meeting in a pecha-kucha format. In this style, each presenter must use 20 slides for 20 seconds each, for a total of presentation time of 6:40 minutes.



Figure 3.3: People gathered for a pecha-kucha presentation.

To tell a story in this fashion, you have to keep your message short and you have to think about how to tell it visually. Allowing 20 minutes for questions and discussion after each presentation, we still cut the meeting to four hours with a 15-minute break at the two-hour mark.

It was a bit of a risk because we didn't know how pecha-kucha would be received. We also had to overcome some pushback from presenters who swore they could never tell their stories in 6:40.

But the executives loved the meeting. They had 20 minutes after each presentation to ask questions and this allowed them to get what they wanted. At the end of the meeting, the executives commended the practice. Our chief learning officer even talked about using the technique in other meetings.

Because I had a trusting relationship with my director and our group encouraged the freedom to question, we were able to talk about and discover the issue and then come up with a simple solution that exceeded customer expectations.

3.3.2 Integrity and credibility

Trust relies on credibility. If people see leaders in the organization say one thing and do another, it undermines credibility. Without actions to back them up, value statements are just words.

In *Software for Your Head*, Jim and Michele McCarthy refer to this quality as integrity [McC02]. Their simple definition of integrity is the unity of thought, word, and deed:

Although that definition may seem abstract, personal integrity is itself an abstract thing. Integrity can be presumed when someone does what he has previously promised to do, or behaves as if he believed in what was said previously. If your actions and words align consistently, you will be judged by others to have integrity. For all practical purposes, if you act as if you have integrity, then you do have integrity.

Most employees can quickly tell you what to believe about their company's mission statement. Often, there are some distinct differences between what the company claims to stand for and the actual culture.

Enron was the extreme example of a company with no integrity. Enron's stated values in its mission statement were respect, integrity, communication, and excellence [Ant15]. Here is how the company defined respect:

We treat others as we would like to be treated ourselves. We do not tolerate abusive or disrespectful treatment. Ruthlessness, callousness, and arrogance don't belong here.

The fact that ruthlessness, callousness, and arrogance are even mentioned should have been a warning sign. As corporate communications editor James Kunen wrote [Kun02], "I've read hundreds of companies' vision and values statements, and nowhere have I seen a single reference to ruthlessness, callousness or arrogance — let alone all three."

Integrity leads to credibility.

Integrity and credibility are also big issues in the branding world. Brands stand for something and if you deliver on this promise, you can be hugely successful. If, however, what you stand for and what you're delivering are two different things, credibility can be an issue:

- Do actions match words?
- Do incentives match rhetoric?

- Are people rewarded for actions that demonstrate values?

In an agile environment, credibility helps business people and developers coordinate. The faster both sides can establish that they mean what they say, the more trust they build, and they can quickly work together to deliver value.

David on the importance of credibility to training and sales

In 1998, Cisco Systems moved into the world of IP telephony with the purchase of Selsius Systems. By 2004, IP telephony was becoming more than just phone calls over an IP network. The vision was absorbing all communication over the Internet: voice, video, and data. The network could be used as the platform for what would become known as “unified communications” (UC). [For05].

Early sales for IP telephony were built on a strong return on investment (ROI) related to bypassing traditional long-distance call charges. As the technology evolved into a more complete communications platform, the sale changed. The benefits of UC became softer benefits related to how customers might do business differently.

For example, time to market for perishable goods is critical in the transportation industry. If drivers find that they can’t reach a destination, shipments must be redirected immediately or lose value. Therefore, drivers need the ability to instantly reach key decision makers who can redirect shipments to alternative destinations. Another example is simply the ability to easily set up WebEx meetings for employees at different locations with different schedules.

If sales teams understood what was most important to customers and how customers currently used communications, they could talk about how UC and collaboration technologies could improve business. This sale is a consultative sell that we also referred to as business transformation. Because the customer sale was no longer based as much on cost saving, our training group was asked to create a business-transformation workshop for sales teams.

Part of the challenge was that this type of a sale, a consultative sale, was quite different from how sales teams had been selling IP telephony. To sell IP telephony, account managers could previously simply pitch return on investment from toll savings to telecom managers. Consultative sales often involved learning more about the business and talking to people outside of the telecom and IT world.

As we put together the workshop and talked with sales teams that had conducted successful business-transformation engagements, one thing became clear: a consultative sales cycle took longer than a technical sales cycle.

Why did this matter?

Because everything sales teams did revolved around weekly commit calls. Account managers had commit goals and they were asked weekly for progress updates. Every person we talked to said that consultative sales were at odds with the commit calls. We risked contradicting ourselves. Management seemed to be saying, “Take more time for a consultative sale,” yet incentives and bonuses were based on weekly commits.

While building the workshop and talking to sales teams, we were advised that we would struggle to convince the sales force if we were to advocate longer sales cycles without eliminating or somehow addressing the weekly commit calls.

We had a couple of ideas for how to resolve the conflict, but as a team we decided we needed to raise the issue to sales leaders. One, we weren’t sure if they fully knew what they were asking and, two, we decided we needed executive backing for our proposals or we knew we’d face fierce opposition.

We reached out to one of the sales vice presidents and told him we did not want to send a mixed message to Cisco sales teams. The vice president said the weekly commit calls weren’t going away anytime soon. We then pitched our idea that account teams selectively target one to two customers for consultative sales. Larger teams that focused on a single enterprise customer wouldn’t have an issue but mid-market account managers could selectively target one to two customers for a longer sales cycle. More time might be invested up front, but there was potential for much larger sales down the road.

The vice president agreed this sounded reasonable and agreed to support the workshop.

In retrospect, we realized the importance of working upfront with account teams. They told us what worked and what issues we might face. As a result, we were able to have an honest conversation with executives and raise issues we’d encountered in order to resolve them in a credible way.

The sales force highly rated the subsequent workshop and business transformation became part of the sales process that continues to this day in Cisco's collaboration group [Cis15].

3.3.3 Craftsmanship

Early agile adoptions focused on empowering people, reducing bureaucracy, improving visibility and collaboration, and adapting to changing requirements. Early agile adoption assumed craftsmanship and that if teams could refocus on getting the bureaucracy out of the way, craftsmanship would shine through.

In reality, some teams lost the focus on technical excellence. As Sandro Mancuso writes in *The Software Craftsman* [Man15]:

Simply adopting Scrum, having daily stand-up meetings, and having tools to manage backlogs and work in progress won't magically improve the quality of the software or make developers better. Improving the process without improving technical excellence is pointless.

This was never the intent of agile, and for this reason, we state craftsmanship explicitly as an aspect of trust so that technical excellence and development skill remain front and center in the agile culture. In any type of complex development environment, whether it's engineering, software, hardware, instructional design, or architectural design, trust is often based on demonstrated skill and quality.

Scientific management has its roots in the industrial revolution and the writings of Frederick Taylor. Taylor believed that people were like interchangeable parts of a machine. He believed the best way to run a company was to have each person doing a job in an exact, prescribed manner so that the company would function like a watch or a mechanical engine. Taylor's model is the assembly line.

How well does this model adapt to complex development? Not very well. Complex development is a creative process and trying to design and develop in assembly-line fashion puts the focus on the process and not the goal of delivering value to the customer. There is also not necessarily one best way to design and develop. Different software developers, for example, could easily come up with different code that does the same thing. Different architects might come up with different building designs that meet a customer's need.

For these reasons, people like Bill Pyritz at Lucent recommend a craftsmanship model for complex development [Pyr03]. Think about any complex development as similar to learning a trade skill like blacksmithing or violin making: developers progress from apprentice to journeymen to craftsmen to master craftsmen.



Figure 3.4: Civil War blacksmith illustration in Harper's Weekly, 1863.

While we don't advocate for these specific roles, we believe that there is benefit to having more senior and skilled members of a team coach more junior members.

Pyritz writes:

For craft teams to work, a commitment is required from management, the craftsmen, and the other members of the team. Most importantly, each team member must truly love his/her work and be skilled in the craft.

Both customers and internal coworkers look for signs of quality and craftsmanship.

Stephen M.R. Covey, in *The Speed of Trust*, writes that we judge competence (or craftsmanship) based on capabilities and results. Capabilities are our talents, our skills, and our knowledge. Results are our accomplishments [Cov06].

If you were interviewing a software developer with a terrific history of coding experience for a management position, you would ask about management capabilities. Similarly, if a person lists only skills, you would wonder what the person has accomplished with those skills. As teams grow and work together, more senior members should look for opportunities to coach junior members and junior members should reach out if there are areas of expertise in which they would like to improve and develop as craftsmen.

Customers look for craftsmanship in organizations to trust that they will receive value:

- What is the organization's definition of quality? Is it different internally than with customers?
- Does the organization meet or exceed customer expectations?
- Does the branding and marketing match the delivery? Does the organization do what it promises (and more)?
- How do coworkers learn about each other's skills and results within the organization?
- What's the relationship between sales and delivery?

By encouraging delivery of small increments, the philosophy of agile can help demonstrate craftsmanship, both within teams and to customers.

3.3.4 Empathy and respect

Empathy and respect are both critical to trust and have a great deal in common. Respect is one of the key feelings people tend to want from others and empathy is the ability to put yourself in another's shoes.

Agile organizations tend to be flat organizations due to the belief that the best designs come from self-organizing teams. One way to consider a self-organizing team is that it has no authority implicitly derived from structures. A better way to reflect on it is that everyone on a self-organizing team has authority.

In this type of environment, respect does not come from hierarchical position but from skill and from showing respect to others. You receive respect when you show respect for others regardless of how they treat you.

In terms of communication, respect often means listening, acknowledging, and discussing rather than attempting to force your own viewpoint. When people don't feel respected, they often respond with anger or blame. Conflict naturally results from differing ideas and different opinions but can be mutually resolved and often leads to better conclusions when there is respect from all sides.

Empathy is the ability to understand other people — not to necessarily agree with them, but to understand them.

U.S. Army Lieutenant General William “Gus” Pagonis, in charge of logistics during the 1991 Gulf War, said [Tur05], “No one is a leader who can't put himself or herself in the other person's shoes.”

Empathy is seeing yourself in another person's situation. The focus is on connectedness, on partnership, and on both of you coming together to accomplish a shared goal. When you can empathize with other people and understand their perspectives, it is much easier to have respect for them.

In agile organizations, empathy and respect help people value individuals and interactions over processes and tools.

Some questions to ask about empathy and respect in organizations are:

- How is respect observed in the organization? What examples of respect and empathy can you cite?
- How are differences viewed in the organization?
- Are people more inwardly focused or outwardly focused? Do they tend to think about others or their own self-preservation?
- What characteristics are respected within the organization?

David on the waiter rule

We invited a team in India new to our learning group at Cisco Systems to dinner at our hotel restaurant. New team members from all over Bangalore met us there.

When I came downstairs from my hotel room, I realized that there were seven restaurants at this particular hotel. I had no idea which restaurant we were meeting at and had to ask. I thought the new team members might be confused as well, so I waited at the hotel entrance to let people know where we were as they were coming in.

We had an excellent meal, enjoyed meeting everyone in a more personal setting, and the onboarding sessions went quite well with everyone engaged in the discussions and activities we'd planned.

Months later, after becoming friends with one of the new team members, I asked about the team's experience and perceptions of the training in Bangalore.

He told me that one of the things they talked about for days was how I had met them at the front door and directed them to where everyone was meeting. He said it made them feel really comfortable and valued, especially as new team members. He told me quite honestly that there was a perception of Americans as self-focused, a perception which I had challenged simply by thinking of them.

I had no idea until months later that the entire team of new hires had discussed this small action and was something they remembered from the training. I hadn't thought twice about it until it was brought up.

Later, I found out that what I'd experienced was an example of the waiter rule: how you treat those in positions of lesser power or authority says more about your character than how you treat those in positions of greater power or authority.

In other words, everyone is going to treat the CEO with respect. Doing so says nothing about your character. What's telling is how you treat others that don't hold any authority over you. Interestingly enough, many CEOs tell similar stories about learning the lesson of the waiter rule [Jon06].



Figure 3.5: The waiter rule. Reprinted with permission. [Mor06]

3.4 Common organizational beliefs about trust

Most people hold similar values. Where they tend to differ is in how they prioritize these values. Their prioritization tends to come from their experiences, their background, their culture, or their religion. Their beliefs, what they hold to be true, influence the importance they attach to different values.

If they've had positive experiences trusting people, they will tend to prioritize trust. If they've had negative experiences, they may rank command and control higher.

Because beliefs and experiences shape values, here are some other beliefs that you want to emphasize and work towards in organizations to build trust:

- People are reliable, self-motivated adults capable of making important decisions.
- People are responsible for their decisions and actions.
- When they understand a goal, people will find the best way to work towards that goal.
- People enjoy work that they are invested in and that fulfills them and helps them develop and grow.
- People want to use their talents to benefit the organization and make a difference.

When projects succeed because of some of these beliefs, discuss how these beliefs lead to success. Again, actions tend to speak louder than words. People will tend to adopt different beliefs when they see leaders adopt them and when they see how they have led to results in situations that they also experience.

Here are some of the beliefs we've heard that might undermine trust:

- People are lazy. If not watched, they will not work diligently [Lal14].
- People work for money [Lal14].
- Demanding something will make it so [Ken Schwaber, in Del15].
- People are selfish and put their interests above those of the organization [Lal14].
- Teams want to run away from work [Ashish Pathak, in Del15].

- People need to be told what to do, when to do it, and how to do it. They also need to be held accountable [Lal14].

The differences are very similar to Douglas McGregor's Theory X and Theory Y.

3.5 Example: How to destroy a high-performing team

David on management culture clashes

One of the stories that I tell about trust centers on my first job as an instructional designer. I was the tenth hire in a small training startup called Horn Interactive. As the new hire, I was given the desk in the elevator hallway as a joke. New people were being hired monthly (so that I could quickly pass on my elevator seat as a rite of initiation) and after a little more than a year we grew to more than 50 people.

As a small team, we grew to know each other quite well. We all sat within shouting distance of each other and if someone wasn't going to work out, everyone knew it well before it happened. Many of us would do things together outside of work, and I'm still in contact with most of the group and friends with several.

Our startup had a couple of differentiators. First, we had people who knew the technical world. I was one of these people. I could explain complex technologies like multiprotocol routers or voice over IP (VoIP) to non-technical audiences. At the time, these technologies were quite new and people questioned whether VoIP could compete with traditional telephony.

Second, we hired extremely talented graphic designers from a renowned local graphic-design program. The e-learning that we developed looked more like video games or comic books or photorealistic simulations than much of the existing cookie-cutter, low-end graphic HTML. To do this high-end development required time and effort and teamwork between the instructional designers, customers, subject-matter experts, and our graphic-design team.

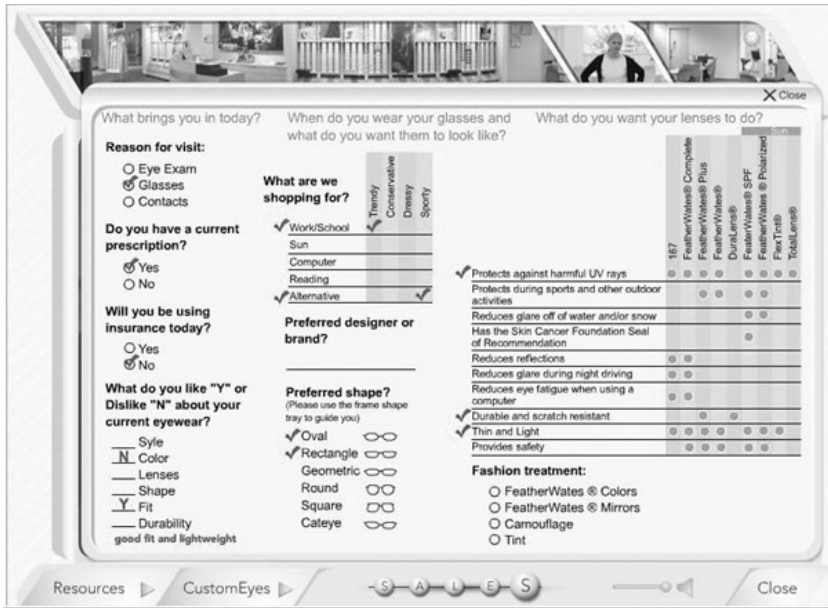


Figure 3.6: Screenshot from photorealistic sales simulation.

The graphic designers, most of whom were fresh out of college, quickly became their own high-performing team. They took over a room in the back of our office, became fast friends, shared much of the same culture, and focused on clients. One of the ways they bonded was through playing video games like Quake. They also had a private mailing list where they would joke around with each other and play Photoshop wars.

As our group grew, one of the people who rose to a management position was an instructional designer who had some unfortunate history with the graphic-design team. He didn't see the value in their play. Instead of seeing these activities as bonding, brainstorming, and letting off steam, he viewed them as "not working". Think about how much more work they could get done if they would simply stop playing Quake.

He decided the graphics team needed a manager who could instill discipline in the group so he hired a former Air Force captain. The graphic designers saw this as bringing in someone to police them. One of the first things the new manager did was institute a morning team status meeting to discuss tasks. The manager wanted to know what was going on and the status of individual projects. The graphic designers saw this meeting as a one-hour waste of time because they knew what was going on and

all the meeting did was take away an hour that they could be spending on graphic design.

The manager insisted on being put on the designers' private mailing list. Of course, they immediately stopped using the list to discuss all things related to their close-knit group. Sometimes this involved issues with other people that they would sort out. Now, they were being scrutinized by management.

As someone who had to work with the graphic-design team, I found that my job became more difficult. Now, there were sides where before there weren't: management, instructional designers, graphic designers, and editors. The graphic designers wanted to know who you sided with — management or them. The answer was really both or neither. The lack of trust had suddenly become an organizational issue. Instead of a single team focusing on the customer, we started to become two teams that required rules and protocols for dealing with each other. And these rules and protocols had consequences in terms of speed and cost.

Some of this dysfunction was related to our growth as a company. Some of the dysfunction came from personal issues between the graphic-design team and the newly appointed manager.

To his credit, the Air Force captain recognized the dynamics of the situation. He didn't want to be a babysitter. He quickly positioned himself for another role in the company and recommended that a graphic designer lead the graphic-design team.

Others within our organization suggested forming more project-focused teams organized around specific customers and projects. These teams consisted of a project manager, instructional designers, graphic designers, and editors.

As these customer-focused teams formed and people again worked closely together on shared customer goals instead of in functional groups, trust grew and we re-established our business around highly performing small teams.

The lesson I learned from this experience was that sometimes self-organizing teams develop their own cultures and their own ways of doing things. Bringing in someone from outside to "manage" an already high-performing team can be perceived as violating trust.

