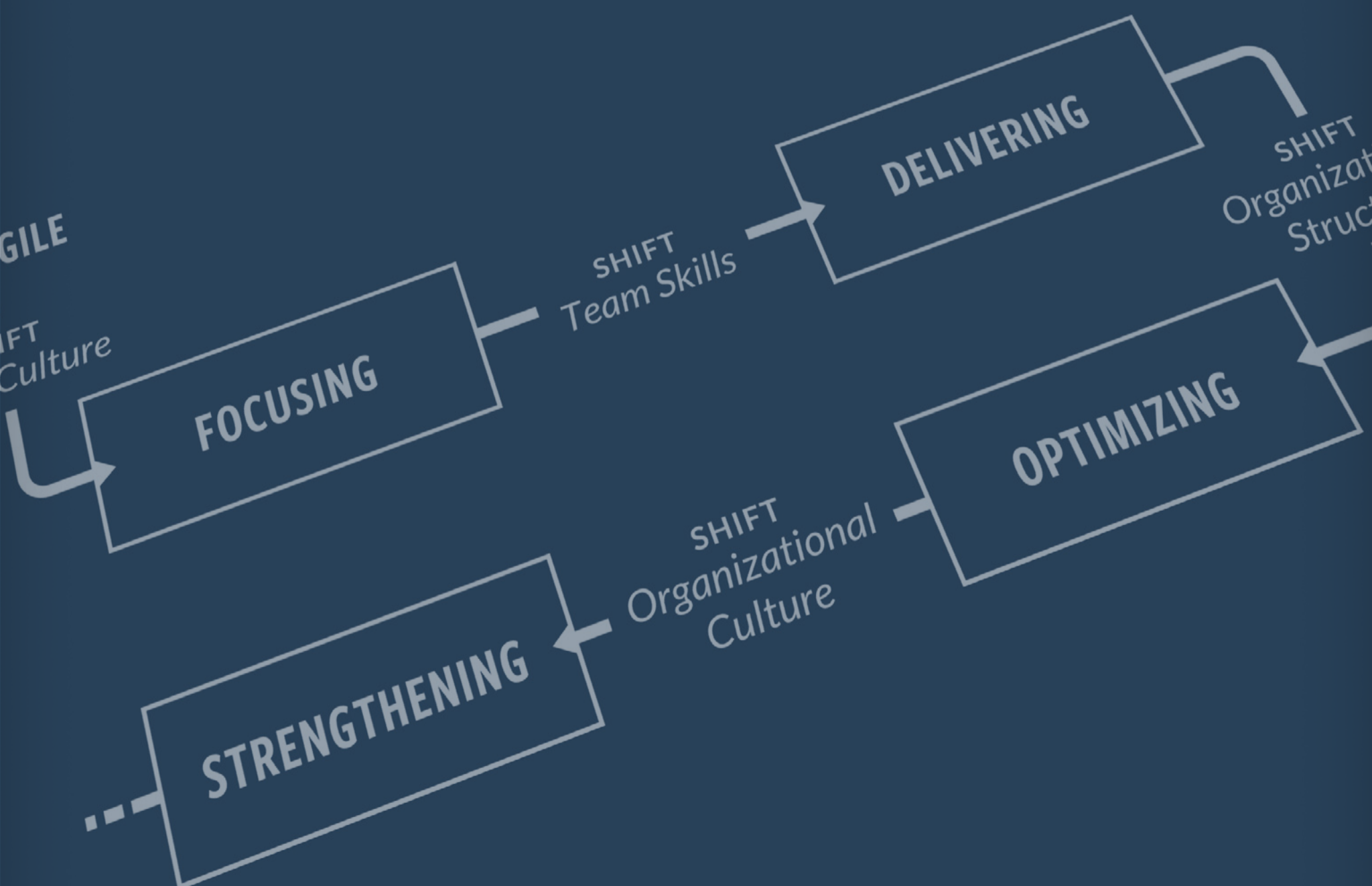DIANA LARSEN AND JAMES SHORE

THE AGILE FLUENCY MODEL

A Brief Guide to Success with Agile

DIANA LARSEN AND JAMES SHORE

THE AGILE FLUENCY MODEL

A Brief Guide to Success with Agile

Contents

A Note from the Authors

We've been leading and helping teams transition to agile development since the late 90's. In that time, we've seen the Agile movement grow from a programmer-centric passion of enthusiasts and innovators to a behemoth that's taken over the software world.

Despite the success—or maybe because of it—agile approaches don't always achieve what people expect. Agile luminaries post articles such as Martin Fowler's Flaccid Scrum (2009). Complaints arise that consultancies profiteer by forcing rigid, un-agile processes down people's throats. Organizational leaders worry that they're not getting the benefits that they should.

> If you'd like to watch a video about this material, check out **this 10-minute overview**.

In our years of helping agile teams, we've learned a lot about what it takes to succeed with agile and why so many organizations don't see the results they expect. In 2012, we formalized this learning as the Agile Fluency™ [1] Model and published it here. In the intervening six years, applying the model has taught us even more. Now we've updated this article to include our latest discoveries.

Use this article to understand which benefits to expect from your agile teams; which investments must be made to achieve those benefits; and where to look when your teams don't deliver the benefits you need. It may involve a mirror.

Introducing the Model

We've observed that agile teams pass through four distinct zones as they learn. Each zone brings specific benefits:

1. **Focusing** teams produce business value.
2. **Delivering** teams deliver on the market cadence.
3. **Optimizing** teams lead their market.
4. **Strengthening** teams make their organizations stronger.

---

1 Agile Fluency is a trademark of James Shore and Diana Larsen. (We've had problems with other people using the term "Agile Fluency" while misrepresenting our model, so we felt we needed to trademark the term to prevent that from happening.)
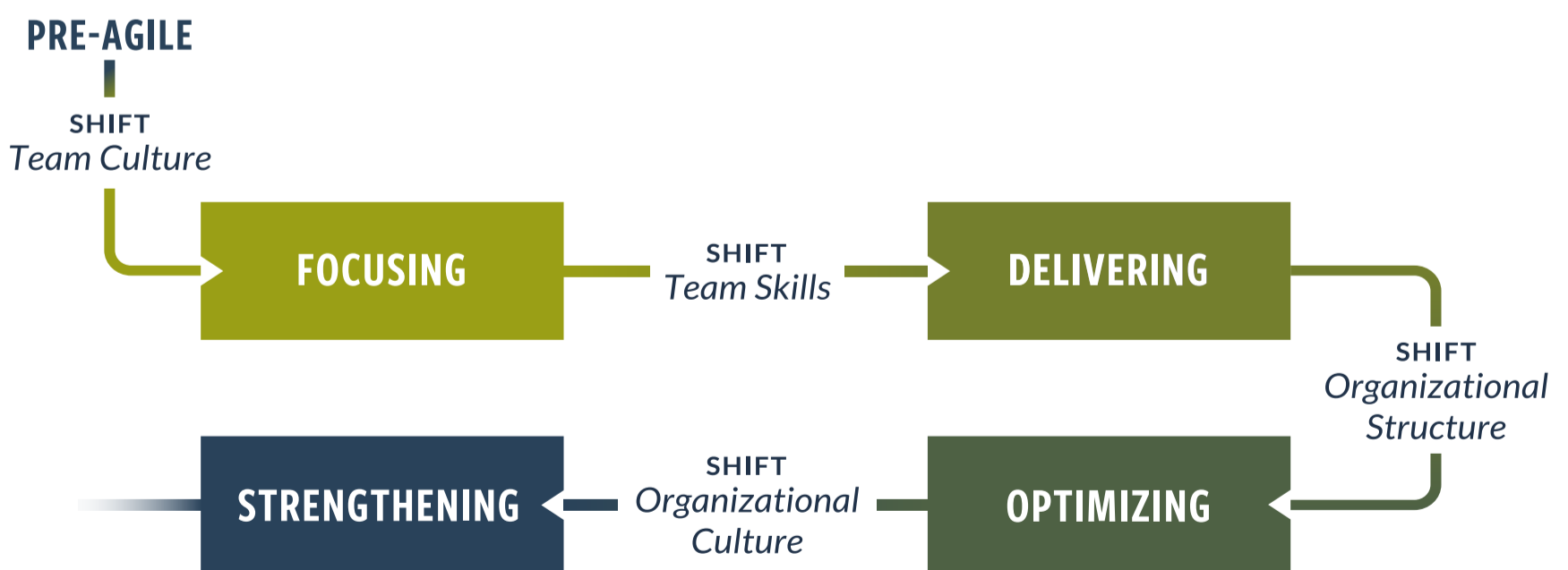
Each zone depends on a set of agile proficiencies. A proficiency is an observable behavior—such as "The team works with a business representative who provides the team with the business's perspective and expectations"—that leads to the zone's benefits.

In the Agile Fluency Model, we're most interested in *fluent* proficiency: a habit of exhibiting the proficiency at all times, even when under pressure.

Agile development is a team sport, so fluency is a trait of the *team*, not individual team members. In practice, some proficiencies will be exhibited by all team members and some will be the specialty of a few individuals. Either way, a team's fluency comes from the ability of team members to self-organize so that individual skills are applied to the right problems at the right times. Anyone can follow a set of techniques when given time to focus in a classroom; true fluency is skillful, routine ease that persists when your mind is distracted with other things.

A team is fluent in a zone when it's fluent in all of the zone's proficiencies, including predecessor zones. Although teams develop proficiencies in any order, even from multiple zones simultaneously, we've observed that teams tend to gain *zone* fluency in a predictable order.

PRE-AGILE

SHIFT
Team Culture

FOCUSING

SHIFT
Team Skills

DELIVERING

SHIFT
Organizational Structure

STRENGTHENING

SHIFT
Organizational Culture

OPTIMIZING

Choose Your Zone

Each fluency zone brings new benefits, so it may seem that the model should be treated as a maturity model, in which the goal is to reach maximum maturity. That would be a mistake. Unlike maturity models, where more mature is always better, the fluency model describes a collection of choices. Each zone represents a fully-mature choice. Each one brings value.

Think of fluency as a ride on a bus. When you get on a bus, you buy a ticket for the zone that you want to reach. A zone that's further away isn't inherently more valuable; it just costs more and takes longer to get to. Sometimes you'll buy a bus ticket for the suburbs, because you want to go to a big box store. Sometimes you'll buy a ticket for downtown, because you want to see a play. Neither is inherently better—it all depends on what you need that day.

Similarly, while each zone has value, each zone also brings challenges. Investing in more than you need could incur organizational backlash, and could even poison people's perception of agile ideas in general.

The appropriate zone for your teams depends on your organization. *Delivering* or *Optimizing* are often the best targets, but *Focusing* and *Strengthening* can also be good choices.

- The *Focusing* zone represents agile fundamentals, and fluent teams provide noticeable benefits to transparency and teamwork. Although *Focusing* fluency doesn't include sustainable technology practices, it's a great way to demonstrate success and create buy-in for further investment. It's also a good fit for teams, such as some digital agencies, that don't maintain their software long-term.

- For teams that need to modify and enhance their software for more than a few months, *Delivering* fluency is often a better choice. This zone represents agile sustainability. *Delivering* teams have low defects, high productivity, and are responsive to business requests. Fluency here is a valuable leap forward for most teams.

- Organizations that wish to set the pace of change in their market, or who see the threat of market disruption on the horizon, will benefit from choosing the **Optimizing** zone. **Optimizing** represents the promise of agile: innovative business agility. Although it has dramatic payoffs, it also requires disruptive changes to organizational structure. Making those changes is often easiest in small, nimble organizations.

- Leaders who want to innovate management theory and practice, particularly in small- to mid-size organizations, may find the **Strengthening** zone to be the best fit for their teams. This zone is a possible future of agile. Cutting-edge agile practice appears to be moving in that direction. However, be cautioned that this zone requires researching cutting-edge management theory and inventing new ways of working.

Even within a single organization, different fluency zones can be appropriate for different teams. Later in this article, we describe the investments and benefits involved for each zone. As you read through the zones, remember that every zone has its own costs and trade-offs. Think carefully about the trade-offs that are right for you rather than simply assuming more is better.

Achieving Fluency

Fluency is more a matter of habits than skills. Although training can teach the underlying techniques, the skillful ease at the heart of *fluent* proficiency requires deliberate, thoughtful, day-in-day-out practice over months. It comes from a deliberate investment in learning through practice.

The proficiencies involved in later zones tend to require more time than the proficiencies for earlier zones. The sooner a team starts working on a zone's proficiencies, the sooner the team will become fluent in them. Agile proficiencies are also mutually supportive. Accordingly, it's best to choose the zone of fluency you want to reach and start practicing all the proficiencies needed for that zone—and all prior zones—simultaneously.

(Of course, you can always change your mind. There's no harm in starting with one zone and targeting a different zone later. It just takes longer.)

As a team practices its proficiencies, fluency will develop in fits and starts. Rather than an orderly progression of proficiency from one zone to the next, proficiencies will develop in parallel across all zones. You're likely to see encouraging signs quickly, but the mastery and reliability of true fluency can be frustratingly slow. Proficiencies plateau, jump forward and back, and progress at different rates.

One of the biggest factors affecting team fluency is organizational support. To continue the bus metaphor, the team has to ride the bus to their fluency zone. No one can do it for them. But the organization has to buy the bus ticket. An organization that expects fluency without providing appropriate support is bound to be disappointed. Even worse, insufficient support can cause turnover and create a cynical corporate culture that hinders improvements. Before embarking on your fluency journey, be sure your organization is prepared to offer the support the journey needs.

One of the biggest investments the organization will make is time. True fluency takes longer than anyone expects or wants. In our experience coaching teams, it takes a team two to six months to become fluent in the *Focusing* zone. Reaching the *Delivering* zone takes another 3–24 months, depending on the amount of technical debt in the code. *Optimizing* fluency can take another 1–5 years, depending on organizational trust and willingness to change reporting structures.

Losing Fluency

It's rare for stable teams to lose fluency on their own. In our experience, it's external disruption that causes fluency loss.

The most common cause of fluency loss is when new management decides that agile approaches don't fit their vision. Without organizational support and the ability to continue practicing what they've learned, team fluency erodes quickly.

This is often accompanied by loss of expertise as dissatisfied team members seek new positions.

Turnover is a related cause of fluency loss. A team that gains or loses too many members may have trouble sustaining what it's learned. This is a particular problem for organizations that assemble new teams for every project.

Rapid growth and imposed processes can also lead to loss of fluency. Successful startups often struggle with this. When they're small, startups often instinctively operate in the *Optimizing* or even *Strengthening* zones. (They're not necessarily fluent in every proficiency, but their instincts push them towards those zones.) Once a startup starts growing rapidly, they often introduce bureaucracy and processes that accidentally damage the ad-hoc processes and culture that enabled fluency.

Similarly, companies that have had success using agile ideas in individual teams sometimes impose a scaling framework on their teams. Most of these frameworks are only designed to support *Focusing* fluency, and some are designed more for management appeal than agile excellence. This makes developing and sustaining fluency at later zones difficult.

That's not to say ad-hoc growth is the correct answer, either. The relationship between growth, scaling, and fluency is a complex topic deserving its own article. For now, we'll just say that, to have fluency at scale, you must have fluency in each individual team. When you evaluate scaling options, consider how your choices support and hinder the proficiencies of the zones you want.

Organizational Fluency

In an agile organization, the work of the organization is done by teams of collaborative people who have a shared purpose and interdependent work. As a result, fluency derives from *teams*. An individual or organization may contribute to a team's fluency, but they can't be fluent on their own.

Although an organization itself can't be fluent, it does make sense to talk about the fluency that an organization *enables*. Team fluency depends on more than just the skills of team members. It also depends on management structures, relationships, organizational culture, and more. Don't make the mistake of blaming individuals for a team's lack of fluency, or assuming that one highly-skilled individual will guarantee fluency. The organizational context often matters more.

When a team hits a roadblock in their fluency development, they'll typically struggle with a few specific proficiencies. Sometimes the problem is a lack of knowledge or skills, and training and mentoring is all the team needs.

More often, the team's development is actually blocked by organizational constraints. You can check for organizational constraints by conducting a fluency diagnostic across several teams. (The Agile Fluency Project offers diagnostic options at agilefluency.org.) If multiple teams are having trouble with the same proficiencies, there's a systemic issue that requires organization-level investment and change.

In the sections that follow, we describe the proficiencies and organizational investments needed for each zone. As you read them, think about which zones your organization is prepared to enable, and which ones it's currently designed to hinder.

Focusing Teams Produce Business Value

Summary . . . . Agile fundamentals
Benefit . . . . Greater visibility into teams' work; ability to redirect.
Investment . . . . . . Team development and work process design.
Learn From . . . . . . Scrum, Kanban, non-technical XP
Time . . . . 2–6 months

Fluent **Focusing** teams work as a cohesive team with shared goals. They think and plan in terms of the benefits their sponsors, customers, and users will see from their software. This is in contrast to teams who are just starting their agile journey, who tend to think in terms of technical considerations, such as software layers, and who often work as individual contributors with individually-assigned tasks.

Focusing: *The team thinks and plans in terms of the benefits their sponsors, customers, and users will see from their software.*

Scrum, Kanban, and the non-technical parts of Extreme Programming are examples of agile methods that teams use to reach **Focusing** fluency. User stories are one of the most common techniques teams adopt. Other techniques include backlogs, retrospectives, timeboxes (such as Sprints), and team task boards. **Focusing** teams also give attention to interaction concepts such as psychological safety, team chartering, group learning, and peer feedback.

Teams fluent in this zone show what they're working on, and how it's progressing from a business value perspective, at least once per month. This is the core metric for **Focusing** teams: it's not the only benefit you should expect from a fluent team, but it's an easy, quick way to check if a team might be fluent. If you don't

have visibility into the team's business priorities, or if those priorities don't reflect what the team's actually working on, then the team isn't yet fluent.

Expected Benefits

TRANSPARENCY

Core Metric	At least once per month, the team shows what it's working on and how it's progressing from a business value perspective.
Reduce Risk	Management knows when the team is building the wrong thing, or isn't making progress, and has the ability to positively intervene.

ACHIEVEMENT

Increase Productivity	The team regularly reflects upon, tunes, and adjusts its work habits to provide more value.
Increase ROI	The team focuses their work on the priority that they are told is most important to the business.
Increase ROI	The team makes incremental progress on business priorities every month.

ALIGNMENT

Increase Productivity	Collaborative communication reduces misunderstandings and hand-off delays among team members.

Focusing benefits derive from effective communication, collaborative teamwork, and continuous team improvement. Fluent proficiency in these categories isn't a technical challenge, but the shift to team culture can be difficult for some people.

Team members must learn to plan in terms of business outcomes rather than technology. They need to learn to take responsibility for the whole team's success, not just their individual contributions. Managers must learn to support teamwork over individual rewards and task planning.

Zone Proficiencies

RESPOND TO BUSINESS NEEDS

The team works with a business representative who provides the team with the business's perspective and expectations.

Business stakeholders can count on the team working on their business representative's view of the next-most valuable thing.

The team plans its work, and shows progress, in chunks that their business representative understands and values.

The team's business representative sees and can change the team's direction at least monthly.

Management enables the team to work at a pace that allows them to respond to business needs indefinitely.

WORK EFFECTIVELY AS A TEAM

The team generates their own day-to-day tasks and plan (based on their business representative's needs).

Team members consider their plan to be the team's work, not individuals' work.

Team members share accountability for getting their plan done.

Management considers the plan to be the team's work rather than assigning accountability to individuals.

> The team embraces, and continuously improves, a joint approach to their work.
>
> The team is aware of how team member relationships affect their ability to succeed and proactively attempts to improve them.
>
> The team is aware of how their work environment affects their ability to do work and proactively attempts to improve it.

Investment/Value Tradeoff: It takes two to six months of practice for a group of independent individual contributors to make the shift to a collaborative, team-based work style. Their fluency depends not only on their efforts, but the investments of their organization. As we've said before, the team may be driving their fluency bus, but their organization needs to buy the bus ticket.

For many managers and organizations, the most challenging investments are non-monetary. Enabling a team to work effectively as a team can require changing management behavior, dedicating team members full-time to their team, and redesigning physical workspaces. In particular, managers must shift from managing the contributions of individuals to managing the work system—guiding team processes, work habits, skills, and context such that the team makes correct decisions without explicit management involvement.

We see many organizations that choose not to make these investments, but still expect fluency from their teams. In these instances, team proficiencies develop slowly and full fluency is rarely achieved. If your organization can't make the requisite investments, agile approaches will disappoint.

Common Organizational Investments

- Select team members with appropriate skills, background, and willingness to work together. Allocate them 100% to their team.

- Create a productivity-focused shared workspace, strongly preferring a physical team room. If a physical team room is infeasible, provide a richly interactive virtual workspace instead, and accept that this will be less effective.

- Ensure that someone with expertise on business priorities and customer value is available to act as the team's business representative.

- Remove impediments to effective teamwork such as competitive ranking, individual rewards, and distributed teams.

- Coach and train team members in Focusing proficiencies.

- Train managers to create an environment that supports teamwork and how to manage the work system rather than individual contributions.

In exchange for making these investments, you'll have greater visibility into what your teams work on and you'll be able to direct them towards the 20% of their work that provides 80% of the value.

Delivering Teams Ship
on the Market Cadence

Summary . . . . Agile sustainability

Benefit . . . . Low defects and high productivity.

Investment . . . . . . Lowered productivity during technical skill development.

Learn From . . . . . . Extreme Programming, DevOps movement

Time . . . . +3–24 months

Fluent **Delivering** teams not only focus on business value, they realize that value by shipping as often as their market will accept it. This is called "shipping on the market's cadence." **Delivering** teams are distinguished from **Focusing** teams not only by their ability to ship, but their ability to ship *at will*.

Delivering: *The team can release their latest work, at minimal risk and cost, whenever the business desires.*

Extreme Programming (XP) pioneered many of the techniques used by **Delivering** teams and it remains a major influence today. Nearly all fluent teams use its major innovations, such as continuous integration, test-driven development, and "merciless" refactoring.

In recent years, the DevOps movement has extended XP's ideas to modern cloud-based environments. The movement's Continuous Delivery and Continuous Deployment techniques are used by most **Delivering** teams. Other useful techniques include evolutionary design, collective ownership, and pair programming or mobbing.

Teams fluent at *Delivering* create low-defect software that can be shipped as often as your organization desires. If a team can't release at will, they aren't yet fluent.

Expected Benefits

TRANSPARENCY

Core Metric	The team can release their latest work, at minimal risk and cost, whenever the business desires.
Reduce Risk	Systemic flaws in your production lifecycle are revealed early.
Increase Satisfaction	The team provides useful release forecasts for managers and customers as needed.

ACHIEVEMENT

Increase Productivity	The team has low defect rates, so less time is wasted fixing bugs and more time is invested in making improvements.
Increase Productivity	The team's codebase has low technical debt, which makes changes cheaper and faster.
Increase ROI	The team ships on the market cadence, capturing value as often as the market will bear.

ALIGNMENT

Increase Productivity	Low defect rates and technical debt are beneficial to job satisfaction and morale, improving retention and productivity.

Delivering is the most technically-intensive fluency zone. There are a lot of skills to learn. Some, such as test-driven development, are of the "moments to learn, lifetime to master" variety. Team members will benefit from studying and practicing techniques described by Extreme Programming, DevOps, and agile software quality gurus. Managers will need to ensure that the team is staffed with people who collectively have all the expertise needed, and they'll need to work with stakeholders to establish an expectation that thoughtful work is valued over expediency.

Zone Proficiencies

RESPOND TO BUSINESS NEEDS

The team's product-related code is production-grade and all its latest work is delivered to a production-equivalent environment at least daily.

The team's business representative may release (or otherwise enable) the team's latest work at will.

The team provides useful release forecast ranges to their business representative upon request.

The team coordinates with their business stakeholders to develop their code and other artifacts in a way that allows them to be maintained, inexpensively, indefinitely.

WORK EFFECTIVELY AS A TEAM

Programmers consider code and similar artifacts to belong to the team, not individuals, and they share responsibility for changing and improving it.

All day-to-day skills needed to design, develop, ship, monitor, maintain, etc., the team's work product are immediately accessible to the team.

When working with code and similar artifacts, team members leave its technical quality at least slightly better than they found it.

Production releases are automated and take no more than ten minutes of manual effort.

The team produces production-grade code without requiring a manual testing phase.

All team members are aware of how their professional and technical skills affect their ability to accomplish the team's goals and decrease their maintenance costs. They proactively seek to improve those skills.

Investment/Value Tradeoff: Developing team members' skills to the point of fluency takes time and significant effort. *Delivering* fluency typically appears 3–24 months after *Focusing* fluency, depending on the amount of coaching the team receives and the amount of technical debt in their codebase. Large systems with very high amounts of technical debt could take even longer.

Training courses can introduce the concepts needed for *Delivering* fluency, but students often have difficulty translating course examples back to their real-world problems. In many cases, fluency also requires engaging skilled practitioner-coaches to work with the team on their real-world projects. In addition, productivity will often appear to decrease as the team learns new skills and pays off technical debt in existing code.

Common Organizational Investments

- Provide time for lowered productivity while team members learn new skills.

- Integrate non-programming technical disciplines, such as QA and ops, into the team.

- Provide training in agile technical practices.

- Engage skilled practitioner-coaches to mentor the team on their real-world work.

Despite the costs, the benefits of **Delivering** fluency are substantial. Fluent teams produce very low-defect software and keep technical debt to a minimum, which means they have more time for delivering features. It takes time for pre-existing technical debt to be paid off and benefits to appear, but once they do, you'll see faster development times, higher quality software, and dramatically improved responsiveness.

Optimizing Teams Lead Their Market

Summary . . . . Agile's promise

Benefit . . . . Higher-value deliveries and better product decisions.

Investment . . . . . . Social capital expended on moving business decisions and expertise into team.

Learn From . . . . . . Lean Software Development, Lean Startup, Beyond Budgeting

Time . . . . +1–5 years

Fluent *Optimizing* teams understand what their market wants, what your business needs, and how to meet those needs. Or, as in a startup environment, they know what they need to learn and how to go about learning it. In contrast to *Delivering* teams, *Optimizing* teams not only have the ability to deliver to market, they also know what to deliver to market.

Optimizing: *The team understands what their market wants, what your business needs, and how to meet those needs.*

Most agile methods are designed to help teams reach *Focusing* or *Delivering* fluency. To gain fluency in the *Optimizing* zone, start with a *Delivering* foundation (such as Scrum + XP + DevOps, Kanban + XP + DevOps, or just plain XP + DevOps) and layer product-centric techniques on top of that foundation.

Lean Startup and Lean Software Development—which, despite the similar names, are different approaches—are both good places to start. Managers will benefit from familiarizing themselves with Beyond Budgeting. From there, be prepared to seek out and experiment with product-focused agile techniques. Useful topics

include customer discovery, continuous product discovery, design thinking, story mapping, and adaptive planning.

Fluent *Optimizing* teams understand their market. They know *why* they're building something, not just what they're building. At a minimum, conversations with a fluent *Optimizing* team will demonstrate a clear-headed view of where their products stand in the market. The team will define its own metrics (or other indicators of progress), they'll be able to defend those choices, and they'll have plans to improve their market position.

If a team doesn't have this sort of understanding of their products and market, or if they produce less value than their opportunity costs, they aren't yet fluent at *Optimizing*. As a corollary, fluent *Optimizing* teams will coordinate with leadership to cancel or pivot low-value products and initiatives.

Expected Benefits

TRANSPARENCY

Core Metric	The team describes where their products stand in their market and how they'll improve their position.
Reduce Risk	The team coordinates with leadership to cancel or pivot low-value projects early.

ACHIEVEMENT

Increase ROI	The team delivers products that meet business objectives and market needs.
Increase ROI	The team learns from market feedback to anticipate customer needs and create new business opportunities.
Increase ROI	The team includes broad-based expertise that promotes optimal cost/value decisions.

Increase Productivity	The team's broad-based expertise eliminates handoffs and speeds decision-making.
Increase Productivity	Mutual trust between the team and its organization leads to rapid, effective negotiation.

One of the biggest challenges in enabling *Optimizing* fluency is giving the team true control over its product direction. The distinction between an *Optimizing* team and a *Delivering* team is that, within the constraints of its charter, the *Optimizing* team makes its own decisions about what to fund and where to focus their efforts. Managers need to delegate this power to teams, which is often a difficult change for organizations.

The distinction between an ***Optimizing*** *team and a* ***Delivering*** *team is that the* ***Optimizing*** *team makes its own decisions about what to fund and where to focus their efforts.*

Of course, to own those decisions, teams need to have the insight to make the *right* decisions... or at least know which experiments will help them discover the right decisions. Gaining that expertise usually involves incorporating non-developers as full-time team members. Common examples include product managers, business analysts, and subject matter experts, but can also include staff from marketing, sales, or customer support.

These sorts of structural changes require high-level permission from the organization. It can be difficult to obtain. Although you may be tempted to hire new employees to fill the gaps, it's usually more effective to include employees who already understand your business's unique priorities and constraints.

Developers and QA personnel, particularly those with company seniority, can be a surprisingly useful source of product expertise. As you're looking for people to

bring business expertise to your team, don't overlook the possibility of training senior technical people in needed business skills. Sales calls and customer visits are a great way to provide new perspectives.

Zone Proficiencies

RESPOND TO BUSINESS NEEDS

The team describes its plans and progress in terms of business metric outcomes jointly identified with management.

The team collaborates with internal and external stakeholders, as appropriate, to determine when and if release forecasts have the best return on investment.

WORK AS A TRUSTED AUTONOMOUS TEAM

The team coordinates with management to understand and refine their role in achieving business strategy.

The team jointly takes responsibility, and accepts accountability, for achieving the business outcomes they identified with management.

Management gives the team the resources and authority they need to autonomously achieve their business outcomes.

Management ensures all day-to-day skills the team needs to understand their market and achieve their business outcomes are embodied in full-time team members.

PURSUE PRODUCT GREATNESS

The team engages with their customers and markets to understand product needs and opportunities.

The team creates hypotheses about business opportunities and conducts experiments to test them.

The team plans and develops their work in a way that allows them to completely change plans, without waste, given less than a month's notice.

Investment/Value Tradeoff: Giving teams decision-making power and corresponding expertise challenges existing organizational structures. It can take several years—not because of the skills required, but because managers and organizational leaders must learn to trust their teams' use of agile ideas before making changes that affect their power, control, and familiar ways of working.

Investing in these changes requires an understanding of positive political skills and a deep conviction in the value of the payoff. Advocates will need to spend their social capital to make it happen. Managers may need coaching in supporting high-performance agile environments, where their job changes from tactical decision-making to defining team direction and orchestrating cross-organization alignment.

Common Organizational Investments

- Dedicate teams 100% to particular products or markets.

- Incorporate business and subject matter experts as full-time team members. Don't assume one person will be enough.

- Give teams responsibility for their budget, plans, and results; judge them on results, not adherence to plans.

- Enable and expect managers to work collaboratively across the organization to remove obstacles to team performance.

- Provide coaching to managers about how high-performance, self-organizing agile teams change the nature of management work.

Optimizing fluency reduces communication overhead, eliminates bureau-
cratic handoffs, and enables rapid response to changing business conditions. Its
investments disrupt the status quo, so they aren't appropriate for every organi-
zation. But for those organizations that wish to drive the pace of change in their
market, an investment in *Optimizing* fluency is worth consideration.

Strengthening Teams Make Their Organizations Stronger

Summary . . . . Agile's future

Benefit . . . . Cross-team learning and better organizational decisions.

Investment . . . . . . Time and risk in developing new approaches to managing the organization.

Learn From . . . . . . Organization design and complexity theories

Time . . . . unknown

Where *Optimizing* teams have the ability to understand and fulfill the needs of their market, *Strengthening* teams also play a larger role in their organization.

Strengthening: *The team understands its role in the larger organizational system and actively works to make that system more successful.*

Strengthening teams contribute to their organization in three ways. First, they understand how they are part of a larger system. They understand how their purpose aligns with other teams' to achieve a greater strategy. They actively work to make that strategy more successful.

Second, they intentionally spread expertise in the organization. They look for opportunities to contribute their skills to other teams, and they seek opportunities to learn from other teams.

Third, the organization distributes directional decisions amongst teams. Leaders design deliberate structures for distilling teams' collective insights and channeling them into organizational improvements.

We see glimmers of *Strengthening* approaches in organizations around the world. Companies like Valve Software, Semco, Zappos, and AppFolio are experimenting in this space. We also see cutting-edge techniques such as team self-selection and Open Space strategy meetings gaining traction in leading agile teams. That said, this zone is speculative. We think it may be the future of agile, but we don't know exactly what it will look like.

Although we aren't entirely certain of the shape this zone will take, we've seen enough examples to draw conclusions about the benefits fluent teams could provide.

Expected Benefits

TRANSPARENCY

Core Metric	The team describes its work in the context of the business's other initiatives, allowing products to be balanced against each other.
Reduce Risk	The team raises, and helps address, cross-organization bottlenecks and issues early.

ACHIEVEMENT

Increase ROI	The team participates in multi-team activities that optimize the organizational value stream.
Increase Productivity	The team recognizes when they can contribute to another team's work, and when that work is more vital, redirects their effort to help them.

Increase Productivity	The team cross-pollinates perspectives, context, learning, and innovations with other teams and other parts of the organization.

This zone is most appropriate for organizations whose leaders want to be on the cutting edge of management theory and practice. It requires working at the leading edge of organizational theory and inventing new ways of applying it to agile teams.

If this zone is for you, research complexity theories, such as Cynefin and Human Systems Dynamics, and new ideas in organization design, including alternative governance structures such as Open Space Agility, Sociocracy, and Holocracy.

Even if you don't desire fluency in this zone, some of the techniques being developed in this space are well worth considering. Just as a fluent **Delivering** team will have some proficiency in **Optimizing** techniques, you can benefit from **Strengthening** techniques. Two we've directly experienced are team self-selection and strategic Open Space sessions. Both are well worth piloting.

For most organizations, **Strengthening** fluency is probably best left as an idea to watch from afar, at least until **Optimizing** fluency is within reach. However, for smaller organizations that already emphasize Lean principles and systems thinking, who are predisposed to distribute decision-making to teams, and who value visionary approaches and innovative processes, the **Strengthening** zone offers a bold challenge and an intriguing puzzle.

Applying the Agile Fluency Model

As George Box said, "All models are wrong, but some models are useful." The Agile Fluency Model is a simplified view of the real world. Despite its simplifications, we've found that it accurately reflects most organizations' needs. Even when it

isn't a perfect match, the benefits, proficiencies, and investments we've described are useful conversation topics.

You can apply the model in three ways: First, use it to see what kind of agile investments your organization needs to make. Insufficient investment not only leads to slow progress, it creates lasting cynicism and resentment. We've seen this failure case too many times. Use the model to make sure your expectations and investments are aligned.

Insufficient investment not only leads to slow progress, it creates lasting cynicism and resentment.

Second, if you aren't seeing the fluency you expect, the model can help reveal what's wrong. Conduct a diagnostic to discover which proficiencies teams are having difficulty with, then provide training and support. (The Agile Fluency Project offers diagnostic options at agilefluency.org.) If multiple teams are having trouble with the same proficiencies, the problem is probably systemic, so look into organizational changes.

Finally, the model is a useful way to align conversations about agile approaches. Discussions about agile ideas can easily get bogged down with arguments about specific methods, tools, and practices. Instead, use the model to foster discussion of what people want to achieve and how they're going to make it possible.

The Agile Fluency Model diagram is available for you to adapt for your presentations. You can also share this article to create conversations about team proficiencies and organizational needs. For derivative works, such as republishing the lists of proficiencies, contact us for permission.

If you need help applying the model or diagnosing fluency challenges, the Agile Fluency Project has contacts and additional resources. Visit agilefluency.org for more.

Conclusion

In our work with agile teams and organizations, we've seen that teams follow a typical progression in their understanding of agile approaches and the benefits their organization receives. We've grouped this progression into four zones of fluency. Each zone is characterized by unique benefits and distinct challenges to adoption.

- The first zone, *Focusing*, requires a team to learn to work together to focus on creating business value rather than merely finishing technical tasks. In return, the organization gains greater insight into the team's work and has more opportunities to influence that work in positive directions. This zone reflects agile fundamentals.

- The second zone, *Delivering*, requires a team to invest in learning a wide array of software development skills. This zone reflects agile sustainability. The skills don't come easily, but with time and adequate organizational support, the team gains the ability to create and ship low-defect software as frequently as the market will accept it, which gives the organization new opportunities for achieving return on their software development investment.

- The third zone, *Optimizing*, represents the promise of agile: a team that dances and turns in response to changing market conditions, and collectively takes responsibility for building the best product your investment can buy. Achieving fluency in this zone means business experts must join the team as full-time contributors, and while this change to organizational structure can be challenging, it pays off in the team's improved ability to serve your business.

- The fourth zone, *Strengthening*, represents agile's future. *Strengthening* teams collaborate with other teams to improve their whole organization. Reaching this zone requires innovative thinking and a willingness to experiment.

All of these zones provide benefits, and every one is the right zone for some teams. Use the model to inspire conversations about which zones your organization wishes to support. Once you've chosen your zones, consider

the investments needed to reach fluency and commit fully to making those investments.

Teams need time to develop their proficiencies. They'll develop in fits and starts, moving forward and backwards, and reaching plateaus. Practice as many proficiencies as you can from the beginning. The fluency zones represent natural plateaus where you'll see distinct benefits and challenges to overcome.

Be aware that organizational context can prevent fluency. Keep an eye out for systemic issues impacting multiple teams in the same way. This is usually a sign that an organizational change needs to be made.

We've seen teams go through these fluency zones time and time again. By sharing our experiences with you, we hope that you'll gain greater insights into the possibilities of agile approaches, and greater understanding of the challenges they pose. May you and your teams achieve ever greater fluency, and ever greater success.

This book is published by the Agile Fluency Project LLC.

Level up to High Performance Coaching in our next Facilitators Workshop!
Learn more and register.

Increase your influence as a trusted advisor.
Become a licensed facilitator.

The Agile Fluency Project also sponsors other workshops and events.
Learn more and register.

Acknowledgements

We want to acknowledge the contributions of so many people. The list to thank includes: reviewers who read our drafts; groups who listened to our presentations; colleagues who challenged our thinking; teams who shared their experiences; and clients who recognized the applicability and utility of our ideas. All offered support and encouragement as we developed the model and then the article. They are many. Mentioning them individually would need another ebook on its own. You know who you are.

We feel particularly indebted to Martin Fowler. He showed his confidence in our work by publishing the original article. He helped to spread the word in many ways. After a few years, he prevailed on us to record what we had learned. This became a revised, updated version published in 2018. He nudged us in just the right way.

We also owe endless gratitude to Adam Light. Adam helped us turn an article and a pile of printouts into a new startup business. He stuck with us long after it stopped making sense.

We appreciate the Agile coaches and consultants everywhere who inspired us. They have used our work to pave the path of the Agile Fluency Suite, then helped businesses and teams along the journey toward agile software development done well. This ebook edition is dedicated to them and their future peers.

Thank you,
Diana and James

About the Authors

The Agile Fluency Project helps coaches, consultants, and change agents put the insights of the Agile Fluency™ Model into practice. We envision a world where Agile is done well in every organization, where all teams practice effectively to sustain continuous improvement, and where each team performs at a level of fluent proficiency that is fit to purpose.
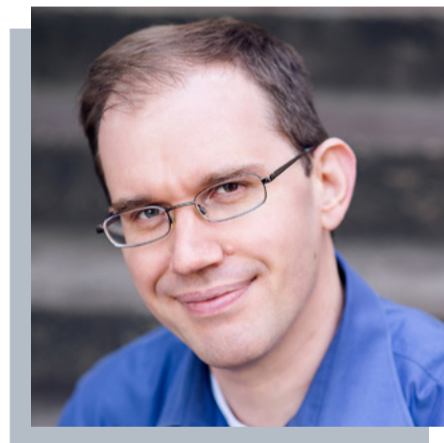
Our Story

James Shore and Diana Larsen fashioned the Agile Fluency Model from their personal observations and deep practical experience. They described the model in collaboration with other thought leaders in the agile community. As an early supporter of the Model, Martin Fowler published their seminal article in 2012.

Following a series of collaborations using the Agile Fluency Model to serve consulting clients, James and Diana co-founded the Agile Fluency Project with Adam Light in 2014.
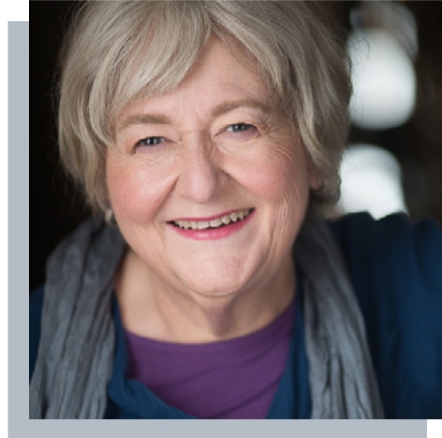
James Shore

James Shore teaches, writes, and consults on Agile development processes. He led his first Agile team in 1999 and was an early adopter of Extreme Programming in 2000. Today, he focuses on helping people understand how the technical, business, and social aspects of Agile fit together to create successful software.

James is a recipient of the Agile Alliance's Gordon Pask Award for Contributions to Agile Practice and co-author of *The Art of Agile Development*. InfoQ has named him as one of the "most influential people in Agile." He writes about software development on his "Art of Agile" blog and hosts the screencast series "Let's Code: Test-Driven JavaScript."

Diana Larsen

Diana Larsen is the author of *Agile Retrospectives: Making Good Teams Great, Liftoff: Start and Sustain Successful Agile Teams,* and *Five Rules for Accelerated Learning*. For more than 20 years she has worked with leaders to design work systems, improve project performance, and support leadership and enterprise agility.

An active speaker and contributor to her professional community, Diana has contributed in leadership roles to the Agile Alliance, the Organization Design Forum, and the Agile Open Initiative.