An Executive’s Guide to Disciplined Agile

Winning the Race to Business Agility
This excerpt contains the following information from the book:

1. List of Chapters
2. Chapter 1: Introduction
3. Index

**List of Chapters**

Preface
1. Introduction
2. Being Disciplined
3. Disciplined Agile Delivery
4. Disciplined DevOps
5. Disciplined Agile IT
6. Disciplined Agile Enterprise
7. A Disciplined Approach to Agile Transformations
8. From Transformation to Continuous Improvement
9. In Conclusion
References and Resources
Acronyms and Abbreviations
Index
About the Authors
Chapter 1: Introduction

Many processes, or methods if you prefer that term, prove to be a prescriptive, one size fits all cookbook. There is often significant marketing rhetoric around the flexibility of the process and how you can tailor it to do anything you want, but in the end they always seem to tell you the “one best way” of doing things and offer very little advice around what your other options are. It is little wonder why so many people are bitter about process these days, yet the hard reality is that we all follow some form of process to do our work. So shouldn’t we try to find a way to tailor and evolve our processes to be as streamlined and effective as possible? That’s what this book is all about.

In many ways Disciplined Agile (DA) is very different than other process offerings. DA is a process decision framework, not a process or methodology. Adoption of DA has really taken off in the last few years, largely because it provides an evolutionary approach towards continuous delivery and DevOps while providing guidance such as agile governance to lend rigor and structure to the overall approach. The approach enables organizations to develop and evolve competitive value streams to provide products and services to their customers. Jon Smart, the Agility Lead of Barclays Group, says it well “We adopted Disciplined Agile across Barclays worldwide because it provides the consistency and structure of a framework with the flexibility of approach at the team level”. In other words, you have the same discipline to live within reasonable governance and constraints that you would expect in a banking situation, but you are free to figure out the details yourselves. Barclays executed the fastest agile transformation of its size seen to date, with currently over 1,000 teams using DA globally.

This chapter is organized into the following topics:

- It’s only getting harder
- The racing car metaphor
- The Disciplined Agile framework
- From scaling to enterprise agile
- Value streams
- Complex adaptive systems
- The (false) promise of bodies of knowledge (BoKs)
- Your improvement journey
- The final destination: Business agility

It’s Only Getting Harder

The need to improve the way that your organization operates, including improving the way that you’re organized and the way that you work, is imperative. It seems that every day there are changes in the business environment, such as new technologies that offer new opportunities or new competitors with business strategies that threaten to disrupt your existing value streams. These external challenges include:

- Customers expect to be delighted, not just satisfied
- Digital transformation, the use of technology to radically improve the way that you serve your customers, is sweeping industry
- The pace of technical change is staggering
- The pace of business change is increasing – Value streams emerge, evolve, and disappear much more rapidly than just a few short years ago
- Overall complexity is increasing as the result of rapid change
- Security threats are increasing and attacks becoming more sophisticated
- Disruptive competitors are blindsiding established organizations and often cutting into the most profitable aspects of their business
- Dramatic political changes are afoot around the world

For example, within the finance industry in particular we have witnessed first-hand a real urgency, indeed panic, by companies to aggressively adopt agile and lean across their organizations. Speaking with their leadership, they universally cite the FinTech revolution as the source of their concern. Recently a new class of company that does not have the cost burden of traditional brick and mortar companies has begun to deliver financial products over the web, collectively known as FinTech companies. They are also applying new technologies such as blockchain to further disrupt the financial industry. In our first book we made the somewhat controversial statement that “those who fail to adopt Agile risk going out of business”. Five years later, we not only continue to believe this is so, but we would go further and suggest that those who fail to evolve their solution delivery capabilities to support the continuous delivery of value to stakeholders risk going out of business.

Of course it doesn’t stop there. Your organization may also be afflicted by many internal challenges, including:
- Organizational structures that reflect long, drawn-out processes
- An aging workforce, a particular problem within IT because much of the knowledge of your core solutions resides in the head of people about to retire
- Organizational silos, each of which have their own priorities and ways of working
- Misalignment between business and IT results in waste and lost opportunities
- Misalignment within IT is common, with various technical silos each having their own “body of knowledge” – more on this later
- Technical debt, quality problems in your systems and overall infrastructure, make it difficult to evolve them quickly to support shifting value streams
- Staff attraction and retention is becoming increasingly harder
- Traditional organizational cultures struggle to address the needs of the new and changing environment you face
- Governance processes based on documentation-heavy, waterfall processes are choking the life out of teams
- Overly specialized staff struggle to evolve and grow with the times
- Workspaces lack flexibility and often inhibit collaboration

One organization that is in the process of adopting a Disciplined Agile approach is the Insurance Corporation of British Columbia (ICBC). Like every other organization that has been in business for many years they suffer from the usual challenges surrounding technical debt and traditional culture. Recognizing the need to optimize the whole organization, not just agile teams, they have restructured their IT division to create long-term stable, cross functional teams aligned to lines of business. They have also redesigned workspaces and where possible collocated their teams. In addition they have been actively investing in training and coaching to help evolve their culture and improve the ways that they work, and they have started into paying down their technical debt in a responsible manner. What makes them stand out, particularly for a North American company, is that they are unionized. Often, the existence of a union increases the friction between workers and management and for the most part decreases the organization’s overall ability to respond in the marketplace. In this case the coaches are working with the union to gain their support to help their members gain agile skills and to fairly rework job descriptions to reflect Disciplined Agile roles.
The Racing Car Metaphor

Since 2001 agilists have been focused on finding ways to improve how software is developed. We’ve adopted fundamental strategies such as regular coordination meetings, regular demos, product owners, developer regression testing, retrospectives, and incremental releases of working software. Disciplined teams have adopted more advanced strategies such as active stakeholder participation, continuous integration (CI), test-driven development, continuous documentation, continuous deployment, measured improvement, and incremental releases of consumable solutions (to name a few). We experiment with new techniques to learn what works for us in the situation that we face, improving our approach as we do so in an incremental kaizen-style manner over time. In effect we are finding ways to tune our “development engines” so that we can deliver more valuable functionality, to reduce our cycle time, and to be more productive as a team. This is very similar to a Formula One team who over the years improves their racing car engine to deliver more power and more speed for less fuel consumption so as to help them win the race.

But agile software development alone isn’t sufficient. We see too many agile teams, who on their own are doing a great job at improving the way that they work, get bogged down by their organizational environment. This is particularly true in established enterprises that have been in operation for decades and sometimes even centuries. The software developers are agile, but you still have a “quality assurance (QA)” group that insists on manual testing based on a detailed requirements document. Or it takes days and sometimes weeks to release a new version of a system into production because of your existing release management practices. You’ve got this great agile team, a great car engine, but you’ve put it into an organizational tractor. Is it any wonder you’re not seeing the desired improvements? Many organizations have come to realize that agile software development alone isn’t enough and now are focusing on DevOps. They’ve increased the scope of their improvement efforts because they realize that their race car engines really need to be put into a race car.

But this isn’t sufficient either. Just like a race car needs a driver and pit crew to operate it, your DevOps strategy is part of your overall IT strategy. If your IT governance approach is based on traditional thinking that requires teams to jump through documentation-oriented “quality gates” then that’s the equivalent of a pit crew putting square wheels onto the car and then holding the driver accountable for losing the race. Similarly your DevOps efforts won’t scale without a flexible, supporting enterprise architecture strategy. Just like your race car requires an effective team to run it in a race, your DevOps strategy requires a supporting IT team and infrastructure for it to be effective.

But this still isn’t enough. A race car and team is of little value if there’s nowhere to race! They are part of a larger racing business which has multiple value streams through which they generate revenue: They make money from ticket sales, from advertisers, from merchandising, and from many other sources. Similarly, your IT department is part of your larger organization, involved with and supporting many value streams from which you make money.

So, to summarize, an engine is part of a race car that is evolved and operated by a team of people and this race car team is part of the overall racing business. Similarly, our software/solution delivery teams are part of an overall DevOps effort, which in turn is part of your IT strategy. Your IT strategy is one aspect of your overall organizational strategy. All of this must work together smoothly, given the challenges described earlier, in order for you to have a truly agile organization. And, on top of that, you need to accomplish this given the myriad of internal and external challenges that you face. How hard could that be?

The Disciplined Agile Framework

Enter the Disciplined Agile (DA) process decision framework. DA provides light-weight guidance to help organizations streamline their information technology (IT) and business processes in a context-sensitive manner. It does this by showing how the various activities such as solution delivery, operations, enterprise architecture, portfolio management, finance, security, legal, and many others
work together. The framework also describes what these activities should address, provides a range of options for doing so, and describes the tradeoffs associated with each option. In effect, DA provides the process foundation for business agility. As you can see in Figure 1.1, there are four levels to the DA framework:

Figure 1.1. The scope of the DA framework.

1. **Disciplined Agile Delivery (DAD).** DAD addresses all aspects of solution delivery, from beginning to end, in a streamlined manner. This includes initial modelling and planning, forming the team, securing funding, continuous architecture, continuous testing, continuous development, and governance all the way through the lifecycle. The framework includes support for multiple delivery lifecycles, including but not limited to a basic/agile lifecycle based on Scrum, a lean lifecycle based on Kanban, two modern agile lifecycles for continuous delivery, and an exploratory lifecycle based on Lean Startup [Ries]. This is the topic of Chapter 3.

2. **Disciplined DevOps.** This is the streamlining of IT solution development and IT operations activities, and supporting enterprise-IT activities, to provide more effective outcomes to an organization. Disciplined DevOps is overviewed in Chapter 4 and an overview of the workflow is shown in Figure 1.2.

3. **Disciplined Agile IT (DAIT).** DAIT addresses how to apply agile and lean strategies to all aspects of IT. This includes IT-level activities such as IT operations, support, data management, reuse engineering, and other capabilities. This is the topic of Chapter 5 and an overview of the workflow is shown in Figure 1.3.

4. **Disciplined Agile Enterprise.** A Disciplined Agile Enterprise is able to anticipate and respond swiftly to changes in the marketplace. It does this through an organizational culture and structure that facilitates change within the context of the situation that it faces. Such organizations require a learning mindset in the mainstream business and underlying lean and agile processes to drive innovation. This is the topic of Chapter 6 and the workflow for which is overviewed in Figure 1.4.
Figure 1.2 The workflow of Disciplined DevOps.
Figure 1.3 The high-level workflow of Disciplined Agile IT (DAIT).
Figure 1.4 The high-level workflow of a Disciplined Agile Enterprise (DAE).
From Scaling to Enterprise Agile

What does it mean to scale agile? The answer to this question depends on who you ask. For example, some people will tell you that scaling agile means applying agile strategies to a large software development team or to a geographically distributed software development team. To others scaling agile means applying agile strategies across a lot of software development teams and to others scaling agile means you apply agile strategies to your organization as a whole. As a result the DA framework distinguishes between two types of “agility at scale”:

1. **Tactical agility at scale (to deal with context).** This is the application of agile and lean strategies on individual Disciplined Agile Delivery (DAD) teams. This includes the ability to apply agile on teams of all sizes, on teams that are geographically distributed, on teams facing regulatory compliance, on teams addressing a complex domain (problem space), on teams applying complex technologies, on teams where outsourcing may be involved, and combinations thereof. An important implication of this is that because you are likely to have delivery teams facing different situations, these teams will be following different tailorings of DAD – context counts.

2. **Strategic agility at scale (to deal with organizational breadth).** This is the broad application of agile and lean strategies across your entire organization. From an IT point of view this includes Disciplined DevOps and Disciplined Agile IT (DAIT) in general. From an enterprise point of view this includes all divisions and teams within your organization, not just your IT department, what we call the Disciplined Agile Enterprise.

You Offer Value Streams to Your Customers

We’ve used the term “value stream” in several places already, so time to define it. A value stream is the sequence of activities an organization undertakes to deliver a product or service to a customer (or a family of related products and services). Most value streams are highly cross functional: the transformation of a customer request or internal idea to a product or service flows through many functional departments or work teams with the organization [MartinOsterling]. As an example, getting a car repaired might include making an appointment, checking in to drop off the car, ordering parts, repairs, pickup, and payment. The value stream touches many cross-functional business processes and possibly multiple IT solutions, in this case to handle appointment scheduling, labor management, customer management, parts inventory, and billing. A value stream may be internal to a company, or it may include external suppliers in addition to the internal processes required to leverage them. A value stream starts and ends with the customer in mind. What problems or needs do they have? What can we do to fulfill them? How can we delight the customer in how we do so? How can we do better? All critical questions you should continuously ask as an organization.

There’s three important points we’d like to make about value streams right now. First, a Disciplined Agile Enterprise is built around value streams, not project streams. Two of the five Disciplined Agile Delivery lifecycles are structured to deliver projects in an agile manner but the framework quite clearly recommends moving away from projects to continuous delivery aligned to value streams. Yes, you may still have projects (although many organizations are now moving away from the concepts of projects, particularly in the IT space) but your primary focus will be on the ongoing efforts around value streams. Second, value streams encompass all the people, resources, and activities required. From the point of view of DA, that implies that a value stream will encompass activities such as solution delivery, operations and support of that solution, product management for the products that support the value stream, governance within the value stream, security, and so on. Yet at the same time there are organizational aspects of these activities that go beyond value streams. Third, value streams have varying lifespans – some only a few months, some a

---

1 This is referred to as the #NoProjects movement.
few years, and some decades or more – that go beyond the normal scope of a project and beyond any annual budgeting processes you may currently have. To properly support value streams you will likely need to improve your organizational processes and structures.

**Your Organization is a Complex Adaptive System**

A Disciplined Agile Enterprise (DAE) is a complex adaptive system. A complex adaptive system is a system in which a perfect understanding of the individual parts does not automatically convey a perfect understanding of the whole system's behavior [CAS]. DAEs are complex because they are a dynamic network of interacting teams, see Figure 1.5, where the overall behavior of the organization is not predicted by the behavior of the individual teams. Having said that, the individual teams are still working towards fulfilling the common goal of the DAE – to delight their customers. When this behavior is positive it's often referred to as synergy, when this behavior is negative it's referred to as a failing organization. DAEs are adaptive because individuals and teams self-organize and learn from their experiences, and hopefully from the experiences of others.

**Figure 1.5. Every team collaborates with and affects other teams.**

This is important because Disciplined Agile thrives when your organization embraces the fact that it is a CAS. Disciplined Agile is about self-organization, improvement, and collaboration amongst other things. Our self-organizing teams will each need to own their process, agile slang for being given process autonomy, and will evolve that process as the team learns from their experiences. Changes in the way that a team works will potentially impact the other teams that it interacts with, those teams will then learn and evolve, which will potentially impact other teams, and so on. The 2016 Agility at Scale study [AoS2016] found that 96% of agile teams reported that they needed to collaborate with one or more groups outside of their team in order to do their work successfully, so this is quite common in practice.

**Uniqueness over Commonality**

Your organization is made up of a collection of interacting teams, each of which follows a process that is unique to them, the team process evolves over time as the team learns, and these teams will have their own specific priorities. Yet, even though your teams are unique they won't be radically different from each other, they will collaborate with one another, overall they will be working towards your organization’s goal(s), and they will still be governed fairly.
Beware The (False) Promise of Bodies of Knowledge (BoKs)

Disciplined Agile Enterprises (DAEs) are complex adaptive systems of interconnected, collaborating teams of people working towards a common goal. An important implication of this interconnectedness is that to be effective you need to improve your overall strategy, not just portions of it. This is very different than the traditional strategy of divide and conquer, the days of allowing teams to locally optimize their processes in isolation through adopting disparate and contradictory “bodies of knowledge (BoKs)” are over. Table 1.1 lists a sampling of the dozens of BoKs available to you, each of which has an organization pushing for its adoption. many of these BoKs contain a lot of “old-style thinking” that is still being presented as leading edge “industry best practices.” There are some great ideas in these BoKs, don’t get us wrong, but those great ideas aren’t always applicable to your situation and these BoKs contain many incredibly questionable ideas. What the DA framework does is it leverages the strategies captured in these BoKs, putting them into context for you so that your teams can choose accordingly – all valuable stuff to capture in your organizational process knowledgebase. To succeed in today’s hyper-competitive environments we must optimize our whole organization, and to do that we must harvest the learnings from a wide range of sources.

Table 1.1. A sampling of bodies of knowledge.

<table>
<thead>
<tr>
<th>Process Area</th>
<th>Bodies of Knowledge (BoKs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Analysis</td>
<td>International Institute of Business Analysis (IIBA) BoK</td>
</tr>
<tr>
<td>Data Management</td>
<td>Data Management Association (DAMA) Data Management BoK (DMBoK)</td>
</tr>
<tr>
<td>Development</td>
<td>The “Agile Canon”, the Software Engineering BoK (SWEBoK), and Software Engineering Method and Theory (SEMAT)</td>
</tr>
<tr>
<td>Enterprise Architecture</td>
<td>The Open Group Architecture Framework (TOGAF), The Zachman Framework, the International Association of Software Architects (IASA) BoK, and the Department of Defense Architecture Framework (DODAF)</td>
</tr>
<tr>
<td>Human Operations</td>
<td>Human Resources Body of Knowledge (HRBoK)</td>
</tr>
<tr>
<td>IT Governance</td>
<td>COBIT (Control Objectives for Information and Related Technology)</td>
</tr>
<tr>
<td>Operations</td>
<td>Information Technology Infrastructure Library (ITIL)</td>
</tr>
<tr>
<td>Product Management</td>
<td>Product Management and Marketing Body of Knowledge (ProdBoK)</td>
</tr>
<tr>
<td>Project Management</td>
<td>Project Management Institute (PMI)’s BoK (PMBoK) and Prince2</td>
</tr>
<tr>
<td>Quality</td>
<td>Various specifications from International Standards Organization (ISO)</td>
</tr>
</tbody>
</table>

Your Improvement Journey

Given the dynamic nature of your organization, a fair question to ask is how can you support the process improvement efforts of your people? We begin with a few key observations:

1. **Improvement is a journey, not a destination.** Real improvement requires years of concerted effort, support, and guidance – not days of training, not weeks of coaching, and not (just) adoption of a new tool.

2. **Many improvement journeys start as a transformation project.** In organizations that still have a project mindset it’s natural to mistakenly assume that you can simply run a short-term transformation project and “viola, you’re now agile!” With a project-based approach you are likely to have many false starts until you finally realize that improvement is a long-term proposition.
3. **Every journey is unique.** Every organization is unique, having its own challenges and its own priorities. You must tailor your approach to reflect the context of the situation that you face, one size does not fit all.

4. **Your goal isn't to be agile, it's to serve your customers better.** Many of your improvements will focus on becoming more agile or lean, but the real goal is to improve your value streams.

5. **Improvement is hard and it requires change.** If improvement were easy we'd all be perfect! Everyone, including managers and executives, must be prepared to adopt a continuous improvement mindset.

6. **Prefer a pull over a push strategy.** When change is forced on people, when you push it on them, they will resist and most likely subvert the change. Change is much more likely to stick when people identify challenges, identify potential improvements, and then willingly choose to experiment with (they pull into their process) the potential improvement(s).

7. **Your improvement efforts need to address people (individuals and interactions), process, and tool issues simultaneously.** Although you'll invest a lot more effort in helping your people, in doing so you'll find that you're simultaneously also helping them to improve their process and to evolve the tooling required to support that process. This idea overviewed in Figure 1.6. Let's work through each of the three improvement aspects in great detail.

---

**Figure 1.6. Relative focus of improvement efforts.**

We have a few observations about how your improvement efforts will affect the people and culture of your organization:

1. **This requires investment in training and coaching.** People need help learning the agile and lean mindset, learning new skills required by the new practices, and learning how to use their new tools properly. This is true for practitioners, for managers, and for executives across your organization.

2. **You can't change people, they have to want to change.** If you want lasting change then people have to believe in that change, they have to want to adopt the new ways of working and thinking. They have to willingly pull changes into their environment and be given the time and resources to adapt to the changes. Otherwise when your “transformation program” ends you'll discover that people slip back into their old, preferred ways of working. Luckily, through training and coaching, people can learn how to improve themselves and their team.

3. **People need to define their own change.** When people are actively involved in identifying improvements, in working through how to implement the improvements, they “own” the change and make it theirs.

4. **People need to be happier with the new environment.** The changes need to make things
better for your staff, they need to be better satisfied with the new way of working otherwise they'll be motivated to change things back to the way they were in “the good old days.”

5. **Some people won't like it, and you'll need to help them.** You will find that some people are perfectly happy with the way that things are, or that they have a very different vision than the one promoted by agile and lean. At best these people will need more help to understand and accept the change over a longer period of time. At worst they will actively fight any effort to become more agile.

6. **Not everyone will survive the journey, and that's ok.** Improvement journeys prove to be perilous for some. When you begin to squeeze the bureaucracy out of your processes the people involved need to find new ways to add value to your organization. Some people are up for that, some are not. Be prepared for some people to push back and even undermine your improvement efforts – expect to have a few unpleasant conversations about your improvement strategy.

7. **Teams own their own processes.** For teams to be effective they need to be in a position to determine and evolve the way that they work. An implication of this is that everyone needs to be flexible enough to interact with other teams in a manner that makes sense to everyone involved.

---

### Improving Your Process

We also have a few observations about how to improve your process:

1. **Recognize that “process” can be a dirty word for many people.** There are several reasons for this: Many existing processes have been pushed onto people; many existing processes are adversarial where people and teams compete for resources; many existing processes are control-based and limit access to information and resources; many processes are inflexible and ill-suited to purpose, many processes are bloated with waste, and many existing processes are coercive in nature in that they provide managers the ability to hire, fire, and punish people. Is it any wonder that many people are leery of process, and in some cases rabidly against it? It doesn't have to be this way. Disciplined agilists seek to build collaborative, flexible, and pragmatic learning processes that empower people to add value to your organization and to delight your customers.

2. **One process size does not fit all.** When every person, every team, and every organization is clearly unique and facing different situations, you must have a process framework that is flexible and easy to tailor. To do this effectively the framework must go beyond the rhetoric of “you can tailor this process to meet your needs” to actually providing real options and making their tradeoffs explicit. DA does this with its goal-driven approach.

3. **Big improvements can occur in small steps.** Small changes, particularly those that move you forward to a larger goal, are less risky, easier to implement, and easier to adopt than large “big bang” changes.

4. **Learn through experimentation.** The safest way to discover what works for you is to actually try out a new idea for a short time, observe and measure how well it works in practice, and then make a fact-based decision as to whether to continue with that change or not. Even if the experiment is a “failure” you’ve still learned something.

5. **Optimize the whole.** Although every organization is a complex adaptive system, and every team owns its own process, they all need to work together as a streamlined whole. For example, if your solution delivery teams are potentially able to release into production on a weekly basis but your release management team insists that they follow a six-week deployment process then you clearly have a problem.

6. **Measure your way to success.** If you don’t measure something it doesn’t improve, the trick of course being that you need to measure what’s truly important. Our advice is to focus on measuring improvements to value streams.

7. **Your improvement targets will evolve over time as your priorities evolve.** As you address your currently pressing challenges you will discover that new opportunities to improve will
You need an organizational process knowledgebase. Just because every team owns its own process that doesn’t imply that they need to learn everything from scratch. Effective organizations enable their team’s process improvement efforts by having supporting libraries of (e-)books, an evolving wiki of practices, commercial products like Enterprise Transformation Advisor from Software Development Experts, references to existing bodies of knowledge, and your corporate policies and guidance.

Keep process documentation very, very light. One of the challenges with all this talk of process improvement is that someone(s) will insist that it needs to be thoroughly documented. Ugh. Few people read process documentation and fewer yet read detailed process documentation. We strongly advise that you take an agile approach and keep your process documentation as lightweight as possible [AgileDocumentation]. One lightweight approach is to have each team define their “collaboration agreements” – sometimes called a team API, team working agreements, rules of engagement, or more formally a service level agreement (SLA). The idea is to define how others can interact with you but to keep the details of how you do so internal to the team.

Improving Your Tooling

There are several observations that we’d like to make regarding your tooling infrastructure:

1. **New ways of working will require new tools.** For software development alone you'll find that you need to adopt a wide array of new tools for testing, continuous integration, and continuous deployment. For governance you'll adopt new dashboard technologies and operational monitoring technologies.

2. **You will use some of your existing tools in new ways.** You may discover that some of your existing tools can be used in a lightweight, more agile manner.

3. **You will need to abandon some of your existing tools.** This won’t be clear to you at first and will likely be painful given the investment you’ve made in them. The strategy is to recognize these investments as the sunk costs that they are and that your real goal is to ensure that your staff has the infrastructure that they need to get the job done.

4. **You'll need to improve your physical work spaces.** This may include building work rooms so that your teams may co-locate, installing whiteboards and other collaborative tooling, and even investing in videoconferencing and smartboard technologies to support geographically distributed work.

   In Chapters 7 and 8 we focus on how to successfully evolve your organization into a Disciplined Agile Enterprise.

The Final Destination: Business Agility

Business agility – an adaptive, lean, responsive, and learning organization – is the true destination of your improvement efforts [Leybourn]. Business agility is something that emerges over time through lots of hard work – there are no shortcuts, silver bullets, or process frameworks that will solve your problems. You must do the work to overcome the challenges that you face. Having said that, you don’t need to go on this journey without a roadmap, nor do you need to do it alone. The Disciplined Agile (DA) framework, as described in this book, is the roadmap that will help get you to business agility. Supported by experienced, certified DA coaches your journey will be faster and less difficult.

Business agility requires true agility across all of IT, not just software development, and a Disciplined Agile Enterprise that is able to leverage that IT capability. And, because the environment in which your organization operates evolves over time, and because your competitors and partners also evolve, business agility proves to be a moving target in practice. We live in interesting times indeed!
Index (From The Book)

#NoProjects, 11

a/b test, 74
activity-based accounting, 133
ADKAR, 149
Agile Data, 98
  database refactoring, 78
agile lifecycle, 47
Agile Manifesto, 21, 32, 33, 149
Agile Marketing Manifesto, 126
Agile Modeling, 37
Agile transformation, 143
  Lean Change, 159
  metrics, 151
  roadmap, 155
  uniqueness, 149
agility at scale, 10, 57
Amazon, 91
analytics, 127
architecture
  and Disciplined DevOps, 79
  and reuse, 109
  and techical debt, 113
  collaborative, 81
architecture owner, 38, 60, 108, 109
audits, 105, 136, 138
automated dashboard, 74, 75, 137
automated deployment, 75
automated regression testing, 74, 82
awesome teams, 24
Barclays, 1, 146
be awesome, 23, 84, 88, 101, 102, 108, 119, 138
Betaworks, 101
beyond budgeting, 103, 111, 132, 157
BizDevOps, 64
BMUF, 98
bodies of knowledge, 13, 88
British Airways, 91
business agility, 19
business intelligence, 66
Business Operations, 141
  and DevOps, 64
business roadmap, 115

canary test, 73
center of excellence, 101, 146, 173
  discontinuing, 174
certification, 150
  snake oil, 180
discontinuing, 101,
change management, 73
chaos monkey, 91
choice is good, 27, 59, 85, 95, 119
  lifecycles, 39
cloud, 83
CMMI, 57, 135
coaching, 15
  benefits, 152
  business, 145
  executives, 144
  internal coaches, 174
  management, 144
  requisite skills, 153
  teams, 145
COBIT, 14
  collaborative architecture, 107
communication plan, 146
community of practice, 101, 148, 171
complex adaptive system, 11, 27, 87, 103, 167
configuration management, 73
consumable solutions, 31, 58
column counts, 25, 52, 59, 85, 92, 106, 119
continuous delivery, 31, 64, 96, 129
continuous delivery agile lifecycle, 49
continuous delivery lean lifecycle, 49
continuous deployment, 74
  separation of duties, 78
Continuous Improvement, 100, 167, 171
  and IT Governance, 106
continuous integration, 49, 74
contracts, 140
Control, 135
  and IT Governance, 104
  and Legal, 131
  and transparency, 138
coordination
  cross-team, 60
cost of delay, 140
culture, 15, 70
  learning culture, 100
  personal safety, 24, 136
Cynefin, 27, 127, 167
DAMA, 13, 97
data governance, 104
Data Management, 66, 97
    and Disciplined DevOps, 77
data quality, 77, 82
data warehouse, 66
database refactoring, 78, 82
delight customers, 22, 84, 92, 119, 128
Deloitte, 130
DevDataOps, 66
development intelligence, 71, 78, 137
DevOps. See Disciplined DevOps
    Engineer, 84
    loop, 63
DevSecOps, 65
disaster planning, 75
disaster simulation, 76
Disciplined Agile
    and complexity, 28
Disciplined Agile Delivery, 6, 37
Disciplined Agile Enterprise, 7, 119
Disciplined Agile IT, 6, 125
    workflow, 87
Disciplined Agile Manifesto, 31, 34
Disciplined DevOps, 6, 63, 90
    and architecture, 79
    anti-patterns, 83
    benefits, 69
    definition, 68
    mindset, 70
    within IT, 88
    workflow, 68, 90
double-loop learning, 33

evacuate waste, 32
emotional intelligence, 153
Enterprise Architecture, 107
    and Disciplined DevOps, 81
    and Reuse Engineering, 108
enterprise awareness, 34, 77, 85, 88, 104, 106, 119
dISCiplinary education, 144
experimentation mindset, 33, 71, 100, 113, 136, 146
exploit testing, 79
exploratory lifecycle, 50, 113
Extreme Programming, 21, 37

fail fast, 31, 58
failure ROI, 100, 169
feature access control, 79
feature toggle, 79
Finance, 131
governance, 104
FinTech, 2
fixed bid, 140
generalizing specialists, 24, 59, 70
geographic distribution, 19
goal diagram
  address changing stakeholder needs, 30
  continuous improvement, 171
  explore initial scope, 53
  release management, 95
goal driven, 52
  mind map, 55
goal question metric, 164
google study, 24
governance, 37, 58
  program, 60
guidance, 81, 137
  data, 98
guild. See community of practice

hackathons, 100
Heart of Agile, 21, 31
Heartland, 96
help desk. See Support
HR. See People Management
HRBoK, 14
hybrid, 39

IASA, 13
IBM, 105
ICBC, 3
IIBA, 13
improvement journey, 14, 167, 179
individuals and interactions, 15
ISO, 14, 57, 135
  27001, 78
IT Governance, 104
  and Control, 104
  and transparency, 138
IT infrastructure, 92
IT intelligence, 71, 78, 137
IT Operations, 67, 91
  and Disciplined DevOps, 75
ITIL, 14, 57
Kanban, 21, 37
knowledge management, 18
KPIs, 138
large teams, 59
Lean Change, 146, 156, 159
lean lifecycle, 47
Lean Software Development, 37
Lean Startup, 21, 50, 113
learn fast, 31, 58
Legal, 130
   and Control, 131
   and People Management, 102
Lego, 125
lifecycle
   agile, 47
   and process improvement, 50
   choices, 39
   continuous delivery agile, 49
   continuous delivery lean, 49
   exploratory, 50, 113
   lean, 47
   risk-value, 58
line of business, 124
long-lived teams. See stable teams
lunch and learns, 100
Management 3.0, 103
Marketing, 126
   and Product Management, 126
McKinsey, 111, 126
metrics, 17, 33
   actionable, 137
   agile transformation, 164
   transformation, 151
microservices, 83
Microsoft, 105, 129
mindset, 35
   continuous improvement, 101
Disciplined DevOps, 70
experimentation, 71, 78, 100, 113, 146
governance, 106
hire for, 102
learn fast, 58
personal safety, 136
venture capital, 132
minimal viable change, 163
mission command, 137
Modern Agile, 21, 31
monitoring, 111
  instrumentation, 79
  security, 79
motivation, 24, 105
moving from/to model, 161
MTBD, 63

Netflix, 91

objectives and key results, 165
offboarding, 101
onboarding, 101
operational intelligence, 78, 137
optimize flow, 31, 50, 59, 85, 94, 96, 97, 115, 119, 121, 130, 138, 170
optimize the whole, 32, 72
organizational learning, 100
organizational maturity, 122
Outside-In Development, 37

PCI DSS, 78, 96
penetration testing, 79
People Management, 101
  and Legal, 102
  staff retention, 70
performance appraisals, 103
personal safety, 24, 136, 176
phases, 39
pilot teams, 145
Pixar, 106
PMBoK, 14
PMI
  ACP, 154
Portfolio Management, 110
pragmatism, 25, 85, 92
principle
  be awesome, 23, 84, 88, 101, 102, 108, 119, 138
  choice is good, 27, 39, 59, 85, 95, 119
  context counts, 25, 52, 85, 92, 106, 119
  delight customers, 22, 84, 92, 119, 128
  enterprise awareness, 34, 77, 85, 88, 104, 106, 119
  optimize flow, 31, 50, 59, 85, 94, 96, 97, 115, 119, 121, 130, 138, 170
  pragmatism, 25, 85, 92
process
  as dirty word, 17
documentation, 18
goal driven, 29, 52
one size fits all, 27, 177
process blades, 87
process improvement
and lifecycles, 50
Procurement, 139
ProdBoK, 14
product flow, 22
Product Management, 115
and Marketing, 126
product owner, 60
and Product Management, 117
product teams. See stable teams
production readiness, 96
Program Management, 59
pull change, 14, 15, 177, 180
purism, 25

quality gates, 4

racing car metaphor, 4
radar chart, 26
refactoring, 82, 113
regulatory compliance, 78, 131, 135
release airport, 77
Release Management, 67, 94
and Disciplined DevOps, 76
release synchronization, 77
release trains, 77
resilience, 71, 80
retrospectives, 171
reuse, 72
and enterprise awareness, 34
Reuse Engineering, 109
and Enterprise Architecture, 108
funding, 110
measuring, 110
reuse engineering team, 109
risk-value lifecycle, 58
roadmap
business, 115
transformation, 175
ROI, 109
roles, 38
rolling wave
budgeting, 111
product planning, 115
technology roadmap, 108
rugged software, 78

SAFe, 30, 39, 95
safety. See personal safety
Sales, 128
sandboxes, 74, 137
for support, 76
scaling, 10
factors, 26, 57
strategic, 10, 57
tactical, 10, 57
scope of DA, 5
Scrum, 22, 30, 37, 52, 169
Security, 65, 96
and Disciplined DevOps, 78
security governance, 104
self organization, 121, 136
self recovery, 80
self testing, 80
self-support, 93
SEMAT, 13
sense & respond, 23
separation of duties, 78
SEPG, 171
skills transfer, 96, 97
small changes, 17
solution monitoring, 75
spider chart, 26
split test, 74
Spotify, 31
stable teams, 33, 49
staff retention, 70
standards, 92
strategic scaling, 10, 57
succeed early, 58
Support, 67, 92
and Disciplined DevOps, 76
SWEBoK, 13
systems design, 23
tactical scaling, 10, 57
talent management. See People Management
Target, 96
teal organizations, 121
team design, 157
technical debt, 72, 81, 113
and architecture, 113
technical stories, 82
technology roadmap, 81, 108
three horizon model, 111, 132
TOGAF, 13
tools, 18
training, 15, 53, 150
  agile, 145
  skills, 146
transparency, 71, 106
  enables governance, 138
  of transformation, 164

Uber, 129
Unified Process, 21, 37, 41
unions, 3
user experience, 92

validated learning, 33, 72
  and Marketing, 126
value streams, 10, 111
  as threads, 123
  cash flow, 132
  vs LOBs, 124
vendor management, 139

work area, 18
work-in-progress, 32

you build it, you run it, 70, 75, 93
  at scale, 92